Numerical analysis of lattice Boltzmann schemes : from fundamental issues to efficient and accurate adaptive methods

Thomas Bellotti

▶ **To cite this version:**

Thomas Bellotti. Numerical analysis of lattice Boltzmann schemes : from fundamental issues to efficient and accurate adaptive methods. Numerical Analysis [cs.NA]. Institut Polytechnique de Paris, 2023. English. NNT : 2023IPPAX041 . tel-04266822

HAL Id: tel-04266822

https://theses.hal.science/tel-04266822v1

Submitted on 31 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2023IPPAX041

Thèse de doctorat

ÉCOLE POLYTECHNIQUE

IP PARIS

Numerical analysis of lattice Boltzmann schemes: from fundamental issues to efficient and accurate adaptive methods

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à l'École polytechnique

École doctorale n°574 École doctorale de mathématiques Hadamard (EDMH)
Spécialité de doctorat : Mathématiques Appliquées

Thèse présentée et soutenue à Palaiseau, le 1er juin 2023, par

THOMAS BELLOTTI

Composition du Jury :

Christophe Chalons Professeur, LMV, Université de Versailles Saint-Quentin-en-Yvelines	Président
Paul Dellar University Lecturer, OCIAM, University of Oxford	Rapporteur
Philippe Helluy Professeur, IRMA, Université de Strasbourg	Rapporteur
Pierre Sagaut Professeur, M2P2, Aix-Marseille Université	Rapporteur
Denise Aregba-Driollet Maître de Conférences, IMB, Bordeaux INP	Examinatrice
Irina Ginzburg Ingénieur de Recherche, MaiAge, INRAE	Examinatrice
Li-Shi Luo Professor, Dep. of Mathematics and Statistics, Old Dominion University	Examinateur
Marc Massot Professeur, CMAP, École polytechnique	Directeur de thèse
François Dubois Professeur, LMSSC, Conservatoire National des Arts et Métiers	Invité

ECOLE DOCTORALE DE MATHEMATIQUES HADAMARD

NNT : 2023IPPAX041

Thèse de doctorat

de l'Institut Polytechnique de Paris

préparée à l'École polytechnique

École doctorale n°574

École doctorale de mathématiques Hadamard (EDMH, ED 574)

Spécialité de doctorat : Mathématiques Appliquées

par

Thomas Bellotti

Numerical analysis of lattice Boltzmann schemes: from fundamental issues to efficient and accurate adaptive methods

Analyse numérique des schémas de Boltzmann sur réseau : des questions fondamentales aux méthodes adaptatives efficientes et précises

Sous la direction de:

Marc Massot	(Professeur, CMAP, École polytechnique)	Directeur de thèse
Benjamin Graille	(Maître de Conférences, LMO, Université Paris-Saclay)	Co-Directeur de thèse
Loïc Gouarin	(Ingénieur de Recherche, CMAP, École polytechnique)	Encadrant

Thèse présentée et soutenue à l'École polytechnique, le 1er juin 2023.
Composition du Jury :

Christophe Chalons	(Professeur, LMV, Université de Versailles Saint-Quentin-en-Yvelines)	Président
Paul Dellar	(University Lecturer, OCIAM, University of Oxford)	Rapporteur, Examinateur
Philippe Helluy	(Professeur, IRMA, Université de Strasbourg)	Rapporteur, Examinateur
Pierre Sagaut	(Professeur, M2P2, Aix-Marseille Université)	Rapporteur, Examinateur
Denise Aregba-Driollet	(Maître de Conférences, IMB, Bordeaux INP)	Examinatrice
Irina Ginzburg	(Ingénieur de Recherche, MaiAge, INRAE)	Examinatrice
Li-Shi Luo	(Professor, Dep. of Mathematics and Statistics, Old Dominion University)	Examinateur
Marc Massot	(Professeur, CMAP, École polytechnique)	Directeur de thèse
François Dubois	(Professeur, LMSSC, Conservatoire National des Arts et Métiers)	Invité

Remerciements

Je tiens d'abord à remercier mes directeurs de thèse Benjamin Graille et Marc Massot pour leur présence déterminante, tant du point de vue scientifique qu'humain. Je me souviendrai en particulier de votre aide en m'ayant posé les bonnes questions au bon moment. Vous avez parfois semé le doute dans mon esprit qui, d'abord troublé, a toujours cherché plus loin en essayant d'abandonner son habituelle paresse. Merci à Loïc Gouarin de m'avoir fait progresser dans ma façon d'écrire du code et d'avoir eu toute la patience nécessaire avec moi. Je peux affirmer, sans peur de me tromper, que ce fut un grand plaisir de travailler avec vous.

En second lieu, je remercie Paul Dellar, Philippe Helluy et Pierre Sagaut d'avoir accepté de rapporter ce long manuscrit de thèse. Vos remarques, questions et conseils, toujours formulés avec bienveillance et rigueur, ont élargi mon regard critique sur mon travail et contribué de façon cruciale à l'amélioration de ce rapport. Un grand merci aux autres membres du jury de thèse : Christophe Chalons, qui en tant que président du jury a permis un déroulement impeccable de ma journée de soutenance ; Denise Aregba-Driollet ; Irina Ginzburg et Li-Shi Luo, qui m'a fait l'honneur de sa présence après un long voyage. Les discussions qui ont suivi mon exposé resteront gravées dans ma mémoire comme un moment de véritable partage scientifique dont j'ai eu l'honneur d'être l'un des protagonistes.

Je souhaite aussi témoigner de ma gratitude à ceux avec qui j'ai pu discuter à divers titres : Christian Tenaud, Laurent Séries, Thierry Magin, Gauthier Wissocq, Stephan Simonis, Francky Luddens, François Dubois, Pierre Lallemand, Stéphane Brull, Romane Hélie, Benjamin Boutin et tant d'autres. Merci aussi à mes collègues doctorants qui ont toujours écouté mes questions parfois bêtes, en particulier Louis, Arthur, Ward, Yoann, Dominik, Apolline, Jessie et Corentin.

Je remercie le personnel administratif du CMAP, en particulier Nasséra Naar, Alexandra Noiret et Nathalie Rodrigues, pour avoir été d'une patience et compréhension infinies à mon égard, permettant le bon déroulement de ma thèse.

Je remercie toute ma famille de m'avoir soutenu dans ces trois années assez spéciales de ma vie. Merci à ma mère Silvia et à mes frères Peter et Martin. Merci à mes grands-parents Daria et Alberto ainsi qu'à mes tantes Laura et Paola. Enfin, le plus grand merci va à mon épouse Charlène, dont le soutient a été indéfectible. Ce modeste travail de recherche lui est dédié, avec tout mon amour.

Résumé général

Le travail faisant l'objet de cette thèse s'inscrit dans le domaine de l'étude des méthodes numériques pour les équations aux dérivées partielles et porte une attention particulière aux schémas de Boltzmann sur réseau. Cette classe de schémas est utilisée depuis la fin des années '80, en particulier en mécanique des fluides, et se caractérise par sa grande rapidité. Cependant, les méthodes de Boltzmann sur réseau sont très gourmandes en termes d'espace mémoire et conçues pour des maillages Cartésiens uniformes. De plus, nous manquons d'outils théoriques généraux qui permettent d'en analyser la consistance, la stabilité et enfin la convergence. Le travail de thèse s'articule autour de deux axes principaux. Le premier consiste à proposer une stratégie permettant d'appliquer les méthodes de Boltzmann sur réseau à des grilles de calcul non-uniformes adaptées dynamiquement en temps, afin de réduire le coût de calcul et de stockage. Le fait de pouvoir contrôler l'erreur commise et d'être en mesure d'employer la méthode quel que soit le schéma de Boltzmann sous-jacent sont des contraintes supplémentaires à prendre en compte. Pour cela, nous proposons d'adapter dynamiquement le réseau ainsi que d'ajuster toute méthode de Boltzmann à des maillages non-uniformes en nous appuyant sur la multirésolution. Cela a permis de proposer un cadre innovant pour des maillages mobiles en respectant les contraintes posées. Ensuite, nous démontrons que la méthode proposée présente d'excellentes propriétés en termes de perturbations introduites sur le schéma originel et qu'elle permet ainsi de réduire les phénomènes parasites liés aux maillages adaptés. L'implémentation de cette procédure dans un logiciel ouvert, permettant de représenter et gérer des grilles adaptées par différentes approches dans un cadre unifié et innovant, est ensuite abordée. Le second axe de recherche consiste à donner un cadre mathématiquement rigoureux aux méthodes de Boltzmann sur réseau, lié en particulier à leur consistance vis-à-vis des EDPs visées, leur stabilité et donc leur convergence. Pour cela, nous proposons une procédure, basée sur des résultats d'algèbre, pour éliminer les moments non-conservés de n'importe quel schéma de Boltzmann sur réseau, en le transformant en un schéma aux différences finies multi-pas sur les moments conservés. Les notions de consistance et stabilité pertinentes pour les méthodes de Boltzmann sur réseau sont donc celles des schémas aux différences finies. En particulier, tous les résultats concernant ces derniers, entre autres le théorème de Lax, se transpose naturellement aux schémas de Boltzmann sur réseau. Une étape ultérieure consiste à étudier la consistance et la stabilité directement sur le schéma de départ sans devoir calculer sa méthode aux différences finies "correspondante". Cela permet d'en obtenir les équations modifiées et de montrer le bien-fondé des analyses de stabilité à la *von Neumann* couramment utilisées au sein de la communauté. Ce nouveau cadre théorique permet aussi d'étudier l'influence de l'initialisation des méthodes sur le résultat des simulations ainsi que d'entamer des études préliminaires sur la monotonie des schémas de Boltzmann sur réseau et sur leurs conditions aux limites, qui constituent des ouvertures pour des travaux futurs.

General abstract

The work presented in this thesis falls within the field tackling the analysis of numerical methods for Partial Differential Equations and pays particular attention to lattice Boltzmann schemes. This class of schemes has been used since the end of the 1980s, particularly in fluid mechanics, and is characterised by its great computational efficiency. However, lattice Boltzmann methods are very demanding in terms of memory space and are designed for uniform Cartesian meshes. Moreover, we lack general theoretical tools allowing us to analyse their consistency, stability and finally convergence. The work of the thesis is articulated around two main axes. The first one consists in proposing a strategy to apply lattice Boltzmann methods to non-uniform grids being adapted in time, in order to reduce the computing and storage costs. The ability to control the error and to be able to use the same approach irrespective of the underlying lattice Boltzmann scheme are additional constraints to be taken into account. To this end, we propose to dynamically adapt the lattice as well as to adjust any Boltzmann method to non-uniform meshes by relying on multiresolution analysis. This allows us to propose an innovative framework for moving meshes while respecting the posed constraints. Then, we demonstrate that the proposed method has excellent properties in terms of the perturbations of the original scheme and that it thus allows to reduce the spurious phenomena linked to the adapted meshes. The implementation of this procedure in an open-source software, allowing to represent and manage adapted grids by different approaches in a unified and innovative framework, is then addressed. The second line of research consists in giving a mathematically rigorous framework to the lattice Boltzmann methods, related in particular to their consistency with respect to the target PDEs, their stability, and thus their convergence. For this purpose, we propose a procedure, based on algebraic results, to eliminate the non-conserved moments of any lattice Boltzmann scheme, by recasting it into a multi-step Finite Difference scheme on the conserved moments. The notions of consistency and stability relevant to lattice Boltzmann methods are therefore those of Finite Difference schemes. In particular, all the results concerning the latter, among others the Lax theorem, are naturally transposed to the lattice Boltzmann schemes. A further step consists in studying the consistency and stability directly on the original scheme without having to calculate its "corresponding" Finite Difference method. This allows us to obtain the modified equations and to show the validity of the *von Neumann* stability analyses commonly used within the community. This new theoretical framework also makes it possible to study the influence of the initialization of the methods on the result of the simulations as well as to initiate preliminary studies on the monotonicity of lattice Boltzmann schemes and on their boundary conditions, which constitute openings for future work.

Contents

Introduction générale

L'évolution d'un grand nombre de systèmes physiques peut se modéliser à l'aide d'équations aux dérivées partielles d'évolution (souvent abrégées par EDPs). Toutefois, exception faite pour un nombre assez limité d'EDPs en général très simples, ces équations ne peuvent pas être résolues explicitement de manière exacte. Même si la détermination de solutions explicites demeure hors de portée, il est tout de même fondamental d'étudier l'existence, l'unicité et la dépendance continue en les données des solutions de ces équations, donc *in fine* leur caractère bien posé au sens de Hadamard. Cela est d'autant plus important que l'on cherchera à approcher la solution de ces EDPs par des méthodes numériques, travaillant sur des discrétisations finies du domaine de définition de ces équations. En effet, il est inutile d'envisager une méthode numérique approchant la solution d'un problème qui est mal posé, par exemple, lorsque sa solution n'est pas unique. Dans le domaine des méthodes numériques pour les EDPs, l'un des principaux enjeux est de pouvoir simuler ces systèmes d'équations par des méthodes à la fois rapides, économes en espace de stockage et dont la convergence vers la solution inconnue—lorsque les pas de discrétisation tendent vers zéro—puisse être établie avec certitude. Pour cela, les méthodes de Boltzmann sur réseau—introduites dans les travaux pionniers de [McNamara and Zanetti, 1988, Higuera and Jiménez, 1989, Higuera et al., 1989]—s'offrent comme une alternative extrêmement rapide aux méthodes traditionnelles (différences finies, volumes finis, éléments finis, *etc.*), en particulier—mais pas seulement—pour la résolution d'équations issues de la mécanique des fluides, comme les équations de Navier-Stokes incompressibles. Il s'agit, en effet, de méthodes explicites en temps, mimant une dynamique mésoscopique basée sur un nombre réduit de vitesses discrètes, auxquelles des densités de "particules" sont associées. À chaque étape de la méthode de Boltzmann sur réseau, ces "particules" effectuent une étape de collision, ou relaxation locale, suivie d'une étape de transport selon leur vitesse respective. La grande rapidité de la méthode vient donc de la localité de la collision, qui permet, entre autres, de paralléliser aisément ces méthodes et du fait que les vitesses discrètes sont choisies de telle sorte à ce que les particules restent "attachées" au maillage discret au cours du temps. Le maillage spatial étant Cartésien uniforme, l'étape de transport peut se résumer à un déplacement de pointeur vers une case mémoire, d'où sa rapidité. Les champs d'application des méthodes de Boltzmann sur réseau sont extrêmement vastes. Sans prétention d'exhaustivité, nous citons la mécanique des fluides avec des écoulements incompressibles [Chen and Doolen, 1998, Lallemand and Luo, 2000], multi-phasiques (voir [Huang et al., 2015a] pour une vision d'ensemble), l'aéro-acoustique [Marié et al., 2009], la magnéto-hydrodynamique [Chen et al., 1991, Martínez et al., 1994, Dellar, 2002, Dellar, 2013b, Baty et al., 2023] et enfin les milieux poreux [Pan et al., 2006]. D'autres types d'équations traitées par la méthode de Boltzmann sur réseau incluent l'équation de transport-diffusion [Zhang et al., 2019], les systèmes hyperboliques de lois de conservation [Graille, 2014, Dubois, 2014] et enfin l'équation de Schrödinger [Zhong et al., 2006]. Le Chapitre 1 vise à présenter les bases ainsi que les notations concernant les schémas de Boltzmann sur réseau, qui seront nécessaires tout au long du manuscrit.

Cependant, les méthodes de Boltzmann sur réseau constituent d'une part une mine d'or et d'autre part une forêt vierge pour les analystes numériciens. En effet, un nombre très important de questions à leur sujet restent ouvertes. Avant d'étayer notre propos, nous remarquons que cela vient—à notre avis—de l'origine algorithmique de la méthode de Boltzmann sur réseau : on conçoit d'abord le schéma (très performant et qui par ailleurs semble donner une bonne solution) et—seulement dans un second temps—on se pose la question de quel jeu d'équations il approche et de quelles sont ses propriétés. Cette origine algorithmique fait que les méthodes de Boltzmann sur réseau utilisent plus d'inconnues que le problème qu'elles visent à résoudre. Ainsi, les schémas de Boltzmann sur réseau sont souvent intrinsèquement "compliqués" du point de vue de leurs propriétés numériques. Nous réfutons l'idée selon laquelle les méthodes de Boltzmann sur réseau marcheraient bien à approcher, par exemple,

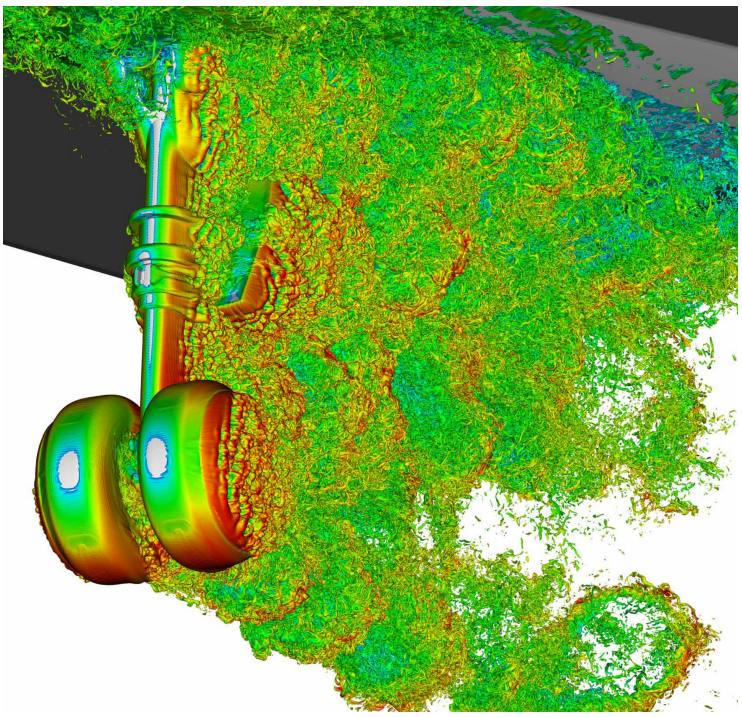


FIGURE 0.1 : Simulation du bruit autour d'un train d'atterrissage en utilisant la méthode de Boltzmann sur réseau avec douze niveaux de grille adaptée. Cela donne un maillage avec 2.28 milliards de cellules et permet une réduction du temps de calcul d'un facteur 15. L'image représente les isolignes du champ de vorticité et le code couleur indique la valeur du nombre de Mach. Courtesy of Michael Barad, Joseph Kocheemoolayil, NASA/Ames (Lattice Boltzmann for Airframe Noise Predictions).

les équations de Navier-Stokes incompressibles, de par leur solide base physique, à savoir, l'équation de Boltzmann [Krüger et al., 2017, Section 2.4]. Alors que—certes—elles peuvent se dériver algorithmiquement à partir d'une version à vitesses discrètes de cette équation, ce qui donne, par ailleurs, leur grande rapidité, elles s'en éloignent sans possibilité de retour à cause du nombre modeste de vitesses discrètes, qui n'est pas augmenté lors de l'étude de convergence. Nous observons que les équations de Boltzmann à vitesse discrète [Broadwell, 1964, Gatignol, 1975, Cabannes et al., 1980, Platkowski and Illner, 1988], tout en conservant qualitativement la structure de l'équation de Boltzmann à vitesse continue, en particulier par des termes de collision non-linéaires quadratiques, n'ont pas comme but d'approcher cette équation. Cette discussion sera approfondie au Chapitre 1.

L'objectif de cette thèse est d'apporter des éléments de réponse à deux problématiques qui affectent les méthodes de Boltzmann sur réseau, décrites dans ce qui suit.

1. La première difficulté est d'appliquer les méthodes de Boltzmann sur réseau à des grilles de calcul non-uniformes. L'objectif d'une telle procédure est de réduire le coût de calcul et surtout de stockage en utilisant moins de ressources aux endroits de l'espace où une résolution fine ne s'avère pas nécessaire. À titre d'exemple, un schéma assez répandu pour traiter des problèmes tridimensionnels, appelé D_3Q_{27} (dans le manuscrit, on indiquera par D_dQ_q tout schéma basé sur un réseau d-dimensionnel avec q vitesses discrètes, en suivant la notation introduite dans [Qian et al., 1992]), nécessite la sauvegarde de vingt-sept quantités à chaque point du maillage. Cela peut s'avérer prohibitif pour des simulations de taille réaliste avec des milliards de cellules, *cf.* Figure 0.1, d'où l'intérêt à diminuer le nombre de mailles. Il ne s'agit donc pas d'une préoccupation secondaire mais d'une nécessité de la part des industriels, voir la NASA [Kiris et al., 2018] et CS Group (ProLB). Cependant, cette nécessité pratique ne doit pas obscurcir les enjeux associés du point de vue de la qualité des solution numériques obtenues. Relever ces défis demande une approche mathématique minutieuse permettant de contrôler l'erreur commise, ce qui n'est pas possible avec les approches heuristiques employées jusqu'à présent, et de minimiser les phénomènes d'ondes parasites introduites par l'adaptation de maillage.

2. La seconde difficulté concerne une compréhension mathématiquement rigoureuse des méthodes de Boltzmann sur réseau dans leur généralité. En particulier, un premier point qui reste à éclaircir est le fait que ces méthodes approchent bien les EDPs visées, donc leur consistance. Une question qui s'ensuit naturellement concerne l'étude de la résilience de ces schémas aux perturbations, donc de leur stabilité. La combinaison de ces deux aspects peut ouvrir la voie à l'étude de la convergence des méthodes de Boltzmann sur réseau vers la solution du problème cible, permettant de garantir la bonne qualité des simulations numériques obtenues. L'objectif ambitieux est donc de faire rentrer les méthodes de Boltzmann sur réseau—dans toute leur généralité—dans le cadre disciplinaire propre à l'analyse numérique, utilisant les outils et concepts spécifiques de cette discipline. Le but est de faire en sorte que les méthodes de Boltzmann sur réseau soient finalement considérées comme des schémas numériques pour les EDPs. Cela permettrait d'impliquer une communauté de chercheurs—disposant de puissants outils d'analyse mais jusqu'à présent plutôt réticente à aborder ce type de problème—dans une redécouverte des schémas de Boltzmann. Cela est d'autant plus intéressant que ces méthodes regorgent de questions fondamentales qui restent ouvertes et qui nécessitent d'angles d'attaque nouveaux. Du point de vue des applications concrètes des méthodes de Boltzmann sur réseau, l'intérêt est double. D'un côté, cela donnera des nouveaux outils d'analyse menant à la conception de schémas avec des meilleures propriétés via-à-vis du modèle physique étudié. De l'autre côté, cela permettra d'expliquer d'un point de vue fondamental un grand nombre de bonnes pratiques et le savoir-faire qui ont été développés "sur le tas" au sein de la communauté d'utilisateurs de la méthode de Boltzmann sur réseau.

Adaptation de maillage pour les méthodes de Boltzmann sur réseau

Renvoyant le lecteur à la partie du manuscrit dédiée à ce premier point pour y trouver un état de l'art plus développé, nous dressons ici un bilan général des études préexistantes, afin de dégager les tendances générales au sein de la communauté. Les deux directions de recherche que nous retrouvons sont les suivantes.

1. Une première consiste à utiliser un pas de temps local à chaque niveau de grille, ce qui demande d'adapter la phase de collision des méthodes afin de préserver les paramètres physiques du problème aux différentes résolutions [Filippova and Hänel, 1998, Lin and Lai, 2000, Kandhai et al., 2000, Yu et al., 2002, Crouse et al., 2003, Dupuis and Chopard, 2003, Rohde et al., 2006, Eitel-Amor et al., 2013, Feldhusen et al., 2016]. La plupart des travaux dans ce courant utilisent des maillages adaptés en espace mais figés en temps. Une étude paradigmatique suivant cette approche, que nous analysons en détail, est celle de [Filippova and Hänel, 1998]. Les auteurs considèrent un schéma D_2Q_9 de type BGK avec trois moments conservés et un maillage grossier sur tout le domaine, auquel se superposent des "patch" à un niveau plus fin, avec un ratio entier n entre pas d'espace par rapport au maillage grossier. Ainsi, le maillage est construit en fonction d'une connaissance préalable du problème. Le paramètre de relaxation est ajusté entre maillage fin et grossier dans le but d'obtenir le même coefficient de viscosité aux différentes résolutions. L'algorithme numérique se déroule en remettant d'abord à l'échelle la solution seulement sur les mailles grossières auxquelles des cellules fines se superposent. Cela fait intervenir les valeurs à l'équilibre, le paramètre de relaxation ainsi que le ratio n. On effectue d'abord une itération (transport et collision) sur tout le maillage grossier, qui ramène ici la solution au temps $t + \Delta t_{\text{grossier}}$. Des interpolations espace-temps d'ordre deux et une formule—faisant intervenir de nouveau les valeurs à l'équilibre, le paramètre de relaxation ainsi que le ratio n—permettent de construire la donnée aux bords des "patch" fins aux temps $t, t + \Delta t_{\text{fin}}, \ldots, t + (n-1)\Delta t_{\text{fin}}$ afin d'effectuer n itérations (transport et collision) sur ces cellules fines. Cela achève un pas de l'algorithme. Cette approche a permis de simuler des allées de *von Kármán* et d'obtenir des valeurs pour les coefficients de drag, de lift et du nombre de Strouhal proches de celles de référence. La seule criticité évoquée dans ce travail concerne le fait que, à cause d'une dépendance du paramètre de relaxation en n, un nombre trop grand de niveaux de grille peut rendre la méthode instable.

2. Une seconde tendance consiste à utiliser un pas de temps global, peu importe le niveau de grille, sans nécessité d'adapter la phase de collision [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016]. Les travaux suivant cette direction s'appuient sur des maillages adaptés dynamiquement en temps en utili-

sant des critères de raffinement propres au problème étudié. Un travail représentatif de cette tendance est celui de [Fakhari and Lee, 2014]. Les auteurs s'attaquent au schéma D_2Q_9 de type MRT avec trois moments conservés et considèrent un maillage adapté en temps avec différents niveaux qui communiquent à l'aide de cellules fantômes. Les valeurs de la solution sur ces cellules sont mises à jour en utilisant des interpolations. Le raffinement du maillage s'appuie sur trois critères : un premier faisant intervenir la vorticité ; un deuxième basé sur la dérivée de la vorticité ; et enfin un troisième basé sur le ratio entre la norme du tenseur des déformations et celle du tenseur de rotation. Le pas de temps est commun à tous les niveaux de grille, sur lesquels une étape de collision est faite sans aucune modification et une étape de transport employant un schéma de type Lax-Wendroff, discrétisant l'équation de transport associée à chaque vitesse discrète. Cette approche est validée sur de nombreux cas test (vortex de Taylor-Green, cavité entraînée, allées de *von Kármán*, *etc.*) avec des résultats qui respectent les valeurs de référence. L'adaptation de maillage permet d'obtenir des facteurs d'accélération du temps de simulation jusqu'à quatre pour certaines configurations. En se comparant aux travaux dans l'esprit de [Filippova and Hänel, 1998], les auteurs de [Fakhari and Lee, 2014] signalent que l'absence d'interpolation temporelle—venant du choix d'un pas de temps global—permet de se passer de remise à l'échelle de la solution et de préserver la précision de la méthode de Boltzmann sur réseau.

Pour résumer, en ce qui concerne la première problématique abordée dans cette thèse, les études disponibles en littérature ont permis d'introduire des manières d'adapter les schémas de Boltzmann sur réseau à des grilles non-uniformes, utilisant soit des maillages fixés au début de la simulation [Filippova and Hänel, 1998, Lin and Lai, 2000, Kandhai et al., 2000, Dupuis and Chopard, 2003, Rohde et al., 2006], soit des maillages adaptés dynamiquement en temps avec des critères prenant en compte le problème considéré [Crouse et al., 2003, Eitel-Amor et al., 2013, Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016, Feldhusen et al., 2016]. Cela permet d'obtenir des gains remarquables en termes de temps de calcul et d'occupation de la mémoire. Cependant, nous observons que les stratégies proposées jusqu'à présent sont rarement adaptatives en temps, ce qui empêche de suivre des phénomènes comme les ondes de choc. De surcroît, les approches disponibles ne permettent pas d'estimer et donc contrôler l'erreur commise en adaptant le maillage. De plus, elles nécessitent—dans la grande majorité des cas—de modifier profondément le schéma sous-jacent ainsi que de définir des critères heuristiques *ad hoc* pour l'adaptation de maillage, d'où le manque de contrôle sur l'erreur commise. Parfois, pire encore, elles demandent de générer un maillage fixe en se basant sur une connaissance *à priori* de l'écoulement étudié. Enfin, l'adaptation de grille est le plus souvent la cause d'ondes parasites aux interfaces entre différentes résolutions [Gendre et al., 2017, Astoul et al., 2021], ce qui pose des difficultés dans beaucoup d'applications, en particulier l'aéro-acoustique.

Ces limites justifient l'étude faisant l'objet de la Partie I de ce manuscrit.

Au Chapitre 2, la démarche consiste d'abord à proposer une approche d'adaptation de maillage dynamique en temps basée sur la multirésolution adaptative [Harten, 1994], ce qui permet de suivre diverses structures—comme les ondes de choc, de détente, *etc.*—présentes dans les solutions et de contrôler l'erreur commise en adaptant la grille de calcul. Ensuite, nous proposons une manière de réécrire des schémas de Boltzmann sur réseau généraux—toujours en utilisant la multirésolution—de façon à ce qu'ils puissent être employés sur ces maillages adaptés en faisant "comme si" nous pouvions dérouler le schéma à la résolution la plus fine, contrairement à l'approche de [Fakhari and Lee, 2014], qui alimente les flux numériques du schéma au niveau le plus fin avec de l'information au niveau de grille local. Tout le procédé étant basé sur la multirésolution, nous prouvons que l'erreur introduite par les méthodes sur grilles adaptées comparée à celle "de référence" sur maillage uniforme est contrôlée. Cette propriété se retrouve dans les nombreuses expériences numériques qui s'ensuivent, accompagnée d'une réduction significative de la trace mémoire des méthodes de Boltzmann sur réseau. Cela montre aussi la généralité de notre manière de procéder, au-delà des spécificités du système considéré. Le contenu de ce chapitre a fait l'objet de deux publications : [Bellotti et al., 2022d] concernant le cadre uni-dimensionnel et [Bellotti et al., 2022c] pour le cadre multi-dimensionnel.

Au Chapitre 3, nous étudions plus en profondeur la méthode proposée. Cela se fait en adaptant l'approche par les équations équivalentes [Dubois, 2008], ce qui permet de caractériser à partir de quel ordre notre stratégie perturbe le schéma de référence. Cela permet donc de montrer que le comportement de notre méthode est au moins d'un ordre plus proche de celui du schéma de référence par rapport à la meilleure approche disponible en littérature [Fakhari and Lee, 2014]. Le point crucial permettant d'obtenir cette "haute-fidelité" par rapport aux approches existantes est de ne pas se borner à calculer les flux numériques au niveau de résolution le plus fin avec des informations venant du niveau de grille local, mais d'employer en plus une reconstruction de la solution au niveau le plus fin afin d'alimenter les flux. Nous montrons par ailleurs que cela a des retombées importantes sur l'amplitude des ondes parasites aux sauts de niveau, qui se trouvent réduites de façon substantielle en utilisant notre approche. Le contenu de ce chapitre a fait l'objet de deux publications [Bellotti et al., 2022b, Bellotti et al., 2022a].

Au Chapitre 4, nous adaptons la technique d'analyse développée au chapitre précédent dans le contexte des schémas de Boltzmann sur réseau afin d'analyser la précision des méthodes volumes finies adaptatives basées sur la multirésolution. Cela permet de quantifier l'ordre de perturbation en fonction de la manière de calculer les flux numériques. Cette information est ensuite intégrée dans l'analyse d'erreur standard pour ce type d'approche et fournit donc une information supplémentaire sur le comportement des schémas, en particulier, en ce qui concerne les solutions réguilières aux endroits du maillage qui ont été déraffinés.

Au-delà de leur conception théorique, nous avons eu besoin d'une implémentation sur ordinateur des méthodes décrites dans la Partie I. Cela fait l'objet du travail présenté dans la Partie II.

Au Chapitre 5, nous présentons les traits saillants de la librairie C++ SAMURAI, qui permet d'aborder les questions d'adaptation de maillage d'un point de vue général, sans se cantonner à la multirésolution par volumes faisant l'objet de cette thèse. Pour cela, nous synthétisons tout maillage Cartésien non-uniforme en regroupant les cellules niveau par niveau et selon leur connectivité spatiale, afin de représenter cela par des intervalles de nombres entiers. Cela donne lieu à un encodage compressé du maillage qui permet d'introduire des opérations algébriques sur les ensembles. Cela s'utilise pour sélectionner des sous-parties du maillage afin d'effectuer des opérations sur celles-ci de manière facile et transparente. Nous faisons un choix judicieux concernant la numérotation des cellules à des fins de stockage, ce qui permet un accès mémoire optimisé. D'ailleurs, le stockage employé s'appuie sur la librairie `xtensor`; afin d'écrire des expressions mathématiques sur les champs stockés associés au maillage en toute simplicité et bénéficier de l'évaluation "paresseuse" desdites expressions.

Au Chapitre 6, nous spécialisons SAMURAI afin d'adapter le maillage *via* la multirésolution et d'implémenter la méthode de Boltzmann sur réseau adaptive correspondante. Cela se fait par une implémentation non-récursive mais itérative, basée non pas sur des arbres de cellules mais sur différentes catégories de cellules. L'implémentation des méthodes numériques conçues durant la thèse a fait l'objet d'un travail qui s'étend sur trois années, en interaction continue avec des développeurs experts, en particulier Loïc Gouarin. Cela s'inscrit dans le cadre de l'initiative HPC@Maths et a permis de fournir à un nombre croissant de chercheurs autour de l'initiative un environnement répondant à leurs besoins concernant l'adaptation de maillage et la multirésolution. Un autre objectif à moyen terme poursuivi au sein de l'initiative est d'intégrer l'adaptation de maillage de SAMURAI dans le logiciel pyLBM, qui permet de représenter simplement des schémas de Boltzmann sur réseau généraux et de les implémenter avec parallélisation par de la génération automatique de code.

Consistance, stabilité et convergence des méthodes de Boltzmann sur réseau

Concernant la seconde grande question abordée dans cette thèse, nous mentionnons que des nombreuses procédures d'analyse de consistance et de stabilité s'appuyant sur des arguments formels existent. Leur caractère formel vient du fait qu'elles donnent des résultats en accord avec les simulations numériques sans se focaliser sur la rigueur mathématique. Pour ce qui est de la consistance, on remarque l'approche "historique" appelée méthode de Chapman-Enskog [Chen and Doolen, 1998, Qian and Zhou, 2000] ; les équations équivalentes [Dubois, 2008, Dubois, 2022] ; l'itération de Maxwell [Yong et al., 2016, Zhao and Yong, 2017] ; ainsi que les analyses asymptotiques de [Junk and Yong, 2003, Junk et al., 2005, Junk and Yang, 2009]. Concernant l'analyse de stabilité, l'approche la plus classique consiste à effectuer une analyse de *von Neumann* sur les schémas linéaires/linéarisés [Benzi et al.,

1992, Sterling and Chen, 1996, Lallemand and Luo, 2000, Graille, 2014, Février, 2014], en vérifiant que les valeurs propres de la matrice du schéma—écrite en utilisant la transformée de Fourier discrète—restent à l'intérieur de cercle unité uniformément en le nombre d'onde. Ces questions feront l'objet d'un état de l'art spécifique dans la Partie III. Au delà de ces travaux, nous essayons maintenant d'identifier deux grands mouvements au sein de la communauté des chercheurs visant des analyses de la consistance et stabilité des schémas de Boltzmann sur réseau avec les outils propres à l'analyse numérique.

1. Un premier angle d'attaque se base sur la formulation originelle des méthodes de Boltzmann sur réseau, dans laquelle les moments conservés (d'intérêt dans les EDPs étudiées) et les moments non-conservés (de nature purement numérique) sont tous présents dans le schéma numérique [Junk and Yang, 2009, Junk and Yang, 2015, Caetano et al., 2023]. Le travail de [Caetano et al., 2023] est un exemple typique de cette tendance. Les auteurs étudient la convergence du schéma $\mathrm{D_1Q_2}$ vers la solution faible entropique d'une loi de conservation scalaire. L'étude est menée dans un régime de sous-relaxation avec donnée initiale à l'équilibre. La première étape clé consiste à prouver un principe du maximum pour le moment conservé et les distributions en vitesse. Cela permet d'obtenir des estimations de la variation totale en temps et en espace, ainsi que de quantifier l'écart du moment non-conservé par rapport à sa valeur à l'équilibre. *In fine*, en combinant ces arguments, la convergence de la solution numérique vers une solution faible—non unique—de l'EDP visée est prouvée. En s'appuyant sur des entropies cinétiques, les auteurs construisent des couples entropies-flux d'entropie numériques et les inégalités d'entropie numériques associées. Une attention particulière est prêtée au fait que ces entropies soient évaluées sur des grandeurs définies après collision, juste avant l'étape de transport. Cela permet de prouver la convergence de la solution numérique vers la seule solution entropique de l'EDP pour la norme L^1. Deux limites, clairement identifiées par les auteurs, sont le fait de ne pas pouvoir étudier le régime de sur-relaxation ainsi que d'aborder l'étude de schémas avec davantage de vitesses discrètes.
2. Un second angle d'attaque procède d'abord à la réécriture du schéma de Boltzmann sur réseau d'origine sous la forme d'un schéma "correspondant", dans lequel les moments conservés restent les seuls présents [Ginzburg, 2009, Suga, 2010, Kuzmin et al., 2011, Lin et al., 2021, Dellacherie, 2014, Fučík and Straka, 2021]. Le travail de [Dellacherie, 2014] est un bon exemple de cette façon de procéder. L'auteur de cette publication considère un schéma $\mathrm{D_1Q_2}$ linéaire. Le point central est un calcul algébrique, effectué en écrivant le schéma numérique originel sur plusieurs pas de temps à différents points de l'espace, qui permet, en recombinant ces expressions, d'éliminer le moment non-conservé. Cela s'achève sur une réécriture du schéma sous la forme d'une méthode aux différences finies multi-pas sur le seul moment conservé. Par conséquent, cela montre rigoureusement que le schéma est consistant avec l'EDP visée et on peut déterminer aisément les conditions de stabilité L^2 et L^∞. Enfin, le théorème de Lax [Lax and Richtmyer, 1956] permet de conclure sur la convergence du schéma—pourvu qu'il soit stable—vers les solutions régulières de l'EDP dans la norme choisie. Une question importante qui reste ouverte est ici identifiée : tenter d'utiliser la même approche pour étudier des schémas de Boltzmann sur réseau plus compliqués ou qui approximent les solutions d'EDPs non-linéaires.

Pour résumer, les travaux antérieurs ont démontré que certains schémas de Boltzmann sur réseau simples ($\mathrm{D_1Q_2}$ [Dellacherie, 2014], $\mathrm{D_1Q_3}$ [Suga, 2010, Lin et al., 2021] et $\mathrm{D}_d\mathrm{Q}_{2W+1}$ TRT [Ginzburg, 2009, Kuzmin et al., 2011], en général avec un seul moment conservé) se réécrivent sous forme de schémas aux différences finies sur le moment conservé, ce qui a permis d'éclairer sur les bonnes notions de consistance, stabilité et convergence de ces méthodes. D'autres travaux [Junk and Yang, 2009, Junk and Yang, 2015, Caetano et al., 2023] agissent directement sur le schéma d'origine, parfois en le comparant [Caetano et al., 2023] aux schémas de relaxation [Jin and Xin, 1995, Aregba-Driollet and Natalini, 2000]. Toutefois, la réécriture sous forme de différences finies ainsi que l'interprétation comme schéma de relaxation ne s'appliquent qu'à des schémas très simples et il n'est pas clair que (et comment) cela puisse se généraliser à tout schéma de Boltzmann sur réseau. Enfin, les méthodes d'analyse formelles mentionnées au début s'appliquent à une vaste gamme de schémas avec un important pouvoir prédictif, mais n'ont pas été intégrées, pour le moment, dans un programme qui pourrait se résumer en "consistance et stabilité impliquent convergence".

Au delà des questions autour de la consistance et la stabilité des méthodes de Boltzmann sur réseau, il est important de prendre en compte le rôle de l'initialisation des schémas ainsi que la présence d'un domaine de

calcul borné, qui demande donc à imposer des conditions aux limites. Pour ce qui est du premier point, l'initialisation des méthodes de Boltzmann sur réseau peut se faire de manière partiellement arbitraire car ces schémas impliquent plus de variables de calcul que de données initiales dans le problème visé. Même si la façon la plus répandue est d'initialiser les variables manquantes à l'équilibre, différents travaux [Caiazzo, 2005, Van Leemput et al., 2009, Huang et al., 2015b] ont proposé des analyses formelles visant à obtenir une compréhension plus fine du comportement des schémas au démarrage en fonction du choix d'initialisation, qui n'est pas forcément à l'équilibre. Ces questions sont très importantes d'un point de vue pratique, en particulier afin de ne pas réduire l'ordre des méthodes à cause de mauvaises initialisations. Concernant les conditions aux limites, l'implémentation pratique des méthodes de Boltzmann sur réseau demande à effectuer les calculs sur un réseau borné. Cela génère un manque d'information lors de la phase de transport aux bords du domaine, qu'il faut combler en imposant une condition aux limites, pour le moment de nature purement numérique. À côté de cela, ces conditions aux limites numériques sur les fonctions de distributions peuvent être employées pour imposer des conditions aux limites physiques (inflow, no-slip, *etc.*) propres au système d'EDPs approchées, exprimées uniquement sur les moments conservés du problème. Cet écart entre conditions aux limites purement numériques et physiques demeure une difficulté majeure au sein de la communauté. De surcroît, les conditions aux limites peuvent introduire des phénomènes d'instabilité numérique qu'il faut arriver à comprendre et à maîtriser. Il existe un très vaste *corpus* de travaux sur les conditions aux bords pour des problèmes et schémas spécifiques, voir [Krüger et al., 2017, Chapter 5]. Nous ne détaillerons pas ici toutes ces contributions. Nous mentionnons juste deux des approches assez répandues. La première est appelée condition de "bounce-back" [Ginzbourg and Adler, 1994, Bouzidi et al., 2001, Dubois et al., 2015] et consiste à combler l'information manquante concernant la distribution d'une vitesse venant de l'extérieur du domaine avec celle de la vitesse opposée. Cela permet—par exemple—d'imposer des conditions "no-slip" sur le champ de vitesse sur des schémas D_2Q_9 avec trois moments conservés. Une seconde condition très utilisée est appelée "anti-bounce-back" [Ginzburg et al., 2008a, Dubois et al., 2020b]. L'information manquante concernant la distribution d'une vitesse venant de l'extérieur du domaine est remplacée par la valeur opposée de celle de la vitesse opposée. Cela permet d'imposer—si utilisée sur un schéma D_2Q_9 avec trois moments conservés—une condition de Dirichlet sur le champ de pression, donc sur la densité. De manière générale, nous manquons d'outils d'analyse des conditions aux limites et cela a des répercussion pratiques importantes, premièrement en termes de fidélité des conditions numériques aux conditions physiques que l'on aurait envie d'imposer et deuxièmement sur la possibilité pratique d'utiliser certaines conditions, à cause de leur instabilité.

Ces questions ouvertes justifient notre étude faisant l'objet des Parties III et IV.

Au Chapitre 7, la démarche consiste à éliminer les moments non-conservés au sein de n'importe quelle méthode de Boltzmann sur réseau à l'aide d'une structure algébrique d'anneau. En effet, ces moments ne sont pas présents dans les EDPs visées. Cela est possible car le théorème de Cayley-Hamilton s'applique aux matrices dont les éléments appartiennent à un anneau commutatif. Une fois les moments non-conservés éliminés, nous restons avec des schémas aux différences finies multi-pas sur les variables conservées, pour lesquels les notions de consistance, stabilité et convergence sont classiques et un grand nombre de résultats sont déjà disponibles en littérature. Le contenu de ce chapitre est inclus dans l'article [Bellotti et al., 2022e].

Au Chapitre 8, nous étudions la consistance des schémas de Boltzmann sur réseau à l'aide des schémas aux différences finies "correspondants". Toutefois, cela s'opère sans écrire explicitement la méthode "correspondante" pour le schéma de Boltzmann sur réseau considéré, mais en caractérisant de manière suffisamment précise la transformation du schéma originel vers le schéma sans les moments non-conservés. Cela est rendu possible par les développements limités de la fonction déterminant et transposée de la comatrice d'une matrice donnée. Ainsi, nous trouvons les équations modifiées de schémas généraux et nous corroborons également les résultats venant de deux techniques formelles d'analyse [Dubois, 2022, Yong et al., 2016]. Le matériel de ce chapitre rentre dans [Bellotti, 2023b].

Au Chapitre 9, nous relions la notion de stabilité à la *von Neumann* pour les schémas aux différences finies multi-pas avec celle pour les schémas de Boltzmann linéarisés. Cette dernière est en effet la plus répandue au sein

de la communauté des schémas de Boltzmann sur réseau. Ce lien est rendu possible—encore une fois—par le fait d'avoir caractérisé la transformation du schéma de Boltzmann sur réseau vers le schéma aux différences finies avec précision. Cela donne un caractère rigoureux à une procédure qui était—jusqu'à présent—utilisée de manière "intuitive".

Au Chapitre 10, nous étudions la question de l'initialisation des schémas de Boltzmann sur réseau, qui a des effets importants sur l'ordre de convergence ainsi que sur la formation de couches limites en temps sur la solution numérique. En effet, cette problématique se pose car les méthodes de Boltzmann sur réseau dans leur formulation originelle présentent plus de variables à initialiser que celles présentes dans l'EDP approchée, alors qu'une fois réécrites comme des schémas aux différences finies, ces schémas sont multi-pas et donc nécessitent des procédures de démarrage. Pour ce faire, nous proposons une analyse basée sur les équations modifiées des schémas près du temps initial, permettant de garantir des initialisations préservant l'ordre des méthodes et d'éviter des oscillations au début des simulations. De plus, en introduisant la notion d'"observabilité" d'un schéma de Boltzmann sur réseau, nous identifions une classe de schémas pour lesquels l'initialisation est facile à étudier avec les outils introduits dans ce travail. Le contenu de ce chapitre fait l'objet d'une pré-publication [Bellotti, 2023a] soumise à une revue à comité de lecture.

Au Chapitre 11, qui constitue un travail préliminaire concernant la stabilité non-linéaire et la monotonie des schémas de Boltzmann sur réseau en utilisant les méthodes aux différences finies correspondantes, nous démontrons la convergence du schéma D_1Q_2 vers la solution faible entropique d'une loi de conservation scalaire en régime de sur-relaxation avec donnée initiale à l'équilibre. L'étude est possible en travaillant sur le schéma aux différences finies "correspondant", en généralisant trivialement la notion de schéma monotone aux méthodes multi-pas. On démontre ainsi un principe du maximum sur le moment conservé, des estimations sur la variation totale en espace et en temps sur le moment conservé. En travaillant avec des couples entropie-flux d'entropie de Krushkov, nous établissons une inégalité d'entropie discrète multi-pas, ce qui permet de conclure. La partie inachevée de ce travail consisterait à étudier le régime de sous-relaxation *via* le schéma aux différences finies afin de retrouver les résultats de [Caetano et al., 2023]. Certains points dans la preuve ont été établis mais d'autres restent ouverts. On peut dire que les propriétés en sous-relaxation se basent fortement sur l'initialisation à l'équilibre, qu'il faut arriver à prendre en compte correctement, et remettent en cause l'universalité d'une généralisation triviale de la monotonie aux méthodes multi-pas [Hundsdorfer et al., 2003].

Enfin, au Chapitre 12, nous entamons un travail préliminaire concernant la consistance et la stabilité des conditions aux limites pour la méthode de Boltzmann sur réseau. On se concentre en particulier sur deux schémas unidimensionnels—à savoir le D_1Q_2 et D_1Q_3—qui peuvent se réécrire, même au bord et pour les conditions aux limites considérées, comme des schémas aux différences finies. Cela permet une analyse de consistance par développement de Taylor et de stabilité en suivant [Gustafsson et al., 1972]. Nous proposons aussi une procédure formelle d'analyse de consistance basée sur l'itération de Maxwell [Yong et al., 2016], qui donne les mêmes résultats que le passage par le schéma aux différences finies "correspondant". Cependant, l'approche passant par l'élimination des moments conservés demeure fortement limitée—lorsque des conditions au bord sont imposées—car nous manquons dans ce cas d'un résultat algébrique général jouant le même rôle que le théorème de Cayley-Hamilton dans le cas d'un domaine non borné/conditions périodiques. Cela peut se voir, même sur les schémas les plus simples, à partir des difficultés à éliminer les moments non-conservés pour des conditions aux limites quelconques.

Publications, présentations et collaborations

Cette thèse de doctorat a donné lieu aux contributions scientifiques suivantes.

Articles dans des revues à comité de lecture

- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022d). Multiresolution-based mesh adaptation and error control for lattice Boltzmann methods with applications to hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 44(4):A2599–A2627

- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022c). Multidimensional fully adaptive lattice Boltzmann methods with error control based on multiresolution analysis. *Journal of Computational Physics*, 471:111670
- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022b). High accuracy analysis of adaptive multiresolution-based lattice Boltzmann schemes via the equivalent equations. *SMAI Journal of Computational Mathematics*, 8:161–199
- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022a). Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps? *Comptes Rendus. Mathématique*, 360:761–769
- Bellotti, T., Graille, B., and Massot, M. (2022e). Finite difference formulation of any lattice Boltzmann scheme. *Numerische Mathematik*, 152:1–40
- Bellotti, T. (2023b). Truncation errors and modified equations for the lattice Boltzmann method via the corresponding Finite Difference schemes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 57(3):1225–1255

Pré-publications et articles en préparation

- Bellotti, T. (2023a). Initialisation from lattice Boltzmann to multi-step Finite Difference methods: modified equations and discrete observability. *Submitted, see https://hal.science/hal-03989355*
- Bellotti, T., Gouarin, L., Leclerc, H., Massot, M., and Séries, L. (2023a). Interval-based data structure for Cartesian meshes: application to multi-scale PDEs on adaptive meshes. *In Preparation*
- Bellotti, T., Massot, J., Massot, M., Séries, L., and Tenaud, C. (2023b). Modified equation and error analyses of adaptive multiresolution Finite Volume schemes. *In Preparation*

Présentations orales

- Initialization from lattice Boltzmann to multi-step Finite Difference methods : modified equations and discrete observability, *invitation 19th International Conference for Mesoscopic Methods in Engineering and Science*, Chengdu (Chine), juillet 2023.
- Initialization from lattice Boltzmann to multi-step Finite Difference methods : modified equations and discrete observability, *22th IACM - CFC 2023*, Cannes (France), avril 2023.
- Finite Difference formulation of any lattice Boltzmann scheme : consistency and stability (and initialization), *invitation au Groupe de Travail d'EDP et Calcul Scientifique du LMRS et LMI*, Rouen (France), février 2023.
- Finite Difference formulation of any lattice Boltzmann scheme : consistency and stability, *invitation au séminaire de l'IRMAR*, Rennes (France), décembre 2022.
- Initialisation pour les schémas de Boltzmann sur réseau : aspects discrets et asymptotiques, *invitation au workshop "Schémas numériques de type Boltzmann"*, Bordeaux (France), novembre 2022.
- Finite Difference formulation of any lattice Boltzmann scheme : consistency and stability, *invitation au séminaire EDP*, Strasbourg (France), octobre 2022.
- Lattice Boltzmann schemes on dynamically adapted meshes relying on multiresolution and recent theoretical breakthroughs in the understanding of lattice Boltzmann methods, *exposé au von Karman Institute*, Bruxelles (Belgique), septembre 2022.
- Limits of the Kinetic Interpretation of Lattice Boltzmann Schemes : A Cure via a Macroscopic Standpoint with Finite Difference Schemes on the Conserved Moments, *32nd Internation Symposium on Rarefied Gas Dynamics*, Séoul (Corée du Sud), juillet 2022.
- Finite Difference formulation of lattice Boltzmann schemes : consequences on consistency and stability, *18th International Conference for Mesoscopic Methods in Engineering and Science*, La Rochelle (France), juin 2022.
- Consistency and stability of lattice Boltzmann schemes, *45ème Congrès d'Analyse Numérique (CANUM)*, Évian-les-Bains (France), juin 2022.

- Finite Difference formulation of any lattice Boltzmann scheme, *invitation au séminaire Analyse Numérique et EDP*, Orsay (France), mars 2022.
- Finite Difference formulation of any lattice Boltzmann scheme, *invitation au 34ème séminaire CEA/GAMNI*, Paris (France), janvier 2022.
- Une formulation de type différences finies pour les scheémas de Boltzmann sur reéseau, *Groupe de travail "Schémas de Boltzmann sur réseau"*, Paris (France), novembre 2021.
- Une formulation de type différences finies pour les scheémas de Boltzmann sur reéseau, *invitation au workshop " Modèles et méthodes pour les équations cinétiques"*, Bordeaux (France), octobre 2021.
- Fully adaptive lattice Boltzmann methods based on multiresolution analysis : error control, high accuracy, equivalent equations and grid jumps, *17th International Conference for Mesoscopic Methods in Engineering and Science*, online, juillet 2021.
- Fully adaptive lattice Boltzmann methods with error control based on multiresolution analysis, *10ème Biennale Française des Mathématiques Appliquées*, La Grande-Motte (France), june 2021.
- Adaptive multiresolution-based lattice Boltzmann schemes and their accuracy analysis *via* the equivalent equations, *Groupe de travail "Schémas de Boltzmann sur réseau"*, Paris (France), mai 2021.
- A class of multidimensional fully adaptive lattice-Boltzmann methods based on multiresolution analysis, *WCCM ECCOMAS 2020*, online, janvier 2021.

Collaborations et interactions

Durant les trois années de thèse, j'ai eu l'occasion d'interagir et collaborer, au-delà des mes encadrants directs (Benjamin Graille, Loïc Gouarin et Marc Massot), avec les chercheurs suivants.

- Laurent Séries, en ce qui concerne le développement de SAMURAI.
- Christian Tenaud, au sujet de la multirésolution et son application aux schémas volumes finis.
- Louis Reboul, pour envisager un lien possible entre schémas de Boltzmann sur réseau et méthodes "asymptotic preserving".
- Thierry Magin, au sujet de l'utilisation des méthodes de Boltzmann sur réseau à des écoulements multiphasiques.

General introduction

The evolution of a large number of physical systems can be modelled using evolution Partial Differential Equations (often abbreviated as PDEs). However, except for a fairly limited number of generally simple PDEs, these equations cannot be solved explicitly in an exact manner. Even if the determination of explicit solutions remains out of reach, it is still fundamental to study the existence, uniqueness, and continuous dependence in data of the solutions of these equations, in short, their well-posed character in the Hadamard's sense. This is all the more important as we will try to approach the solution of these PDEs by numerical methods, working on finite discrete determinations of the domain of definition of these equations. Indeed, it is useless to consider a numerical method approaching the solution of a problem which is ill-posed, for example, when its solution is not unique. In the field of numerical methods for PDEs, one of the main challenges is to be able to simulate these systems of equations by methods that are both fast, economical in storage space, and which convergence to the unknown solution—when the discretization steps tend to zero—can be established with certainty. For this purpose, lattice Boltzmann methods—introduced in the pioneering work of [McNamara and Zanetti, 1988, Higuera and Jiménez, 1989, Higuera et al., 1989]—offer an extremely fast alternative to the traditional methods (Finite Differences, Finite Volumes, Finite Elements, *etc.*), in particular—but not only—for the solution of equations from fluid mechanics, such as the incompressible Navier-Stokes equations. These are, in fact, time-explicit methods, mimicking mesoscopic dynamics based on a reduced number of discrete velocities, to which densities of "particles" are associated. At each step of the lattice Boltzmann method, these "particles" undergo a collision step, or local relaxation, followed by a transport step according to their respective velocity. The efficiency of the method is therefore due to the locality of the collision, which allows, among other things, the easy parallelization of these methods, and to the fact that the discrete velocities are chosen in such a way that the particles remain "attached" to the discrete mesh over time. Since the spatial mesh is uniform, the transport step can be summarised as a pointer displacement to a memory cell, hence its speed. The fields of application of lattice Boltzmann methods are extremely vast. Without pretending to be exhaustive, we cite fluid mechanics with incompressible flows [Chen and Doolen, 1998, Lallemand and Luo, 2000], multi-phase flows (see [Huang et al., 2015a] for an overview), aeroacoustics [Marié et al., 2009], magnetohydrodynamics [Chen et al., 1991, Martínez et al., 1994, Dellar, 2002, Dellar, 2013b, Baty et al., 2023] and finally, porous media [Pan et al., 2006]. Other types of equations treated by the lattice Boltzmann method include the transport-diffusion equation [Zhang et al., 2019], hyperbolic systems of conservation laws [Graille, 2014, Dubois, 2014], and the Schrödinger equation [Zhong et al., 2006]. Chapter 1 aims at presenting the basis as well as the notations concerning lattice Boltzmann schemes, which will be necessary throughout the manuscript.

However, lattice Boltzmann methods constitute—on the one hand—a gold mine and—on the other hand—a virgin forest for numerical analysts. Indeed, a very large number of questions about them remain open. Before elaborating on this, we note that this is due—in our opinion—to the algorithmic origin of the lattice Boltzmann method: one first conceives of the scheme (which is very efficient and which, moreover, seems to give a good solution) and—only in a second step—asks the question of which set of equations it approaches and what are its properties. This algorithmic origin means that lattice Boltzmann methods use more unknowns than the problems they aim at solving. Thus, lattice Boltzmann schemes are often inherently complicated in terms of their numerical properties. We reject the idea that the lattice Boltzmann methods would work well to approximate, for example, the incompressible Navier-Stokes equations, thanks to their strong physical basis, namely, the Boltzmann equation quoted in [Krüger et al., 2017, Section 2.4]. While—admittedly—they can be algorithmically derived from a discrete velocity version of this equation, which gives, incidentally, their impressive performance, they depart from it without any possibility of return because of the modest number of discrete velocities, which is not increased

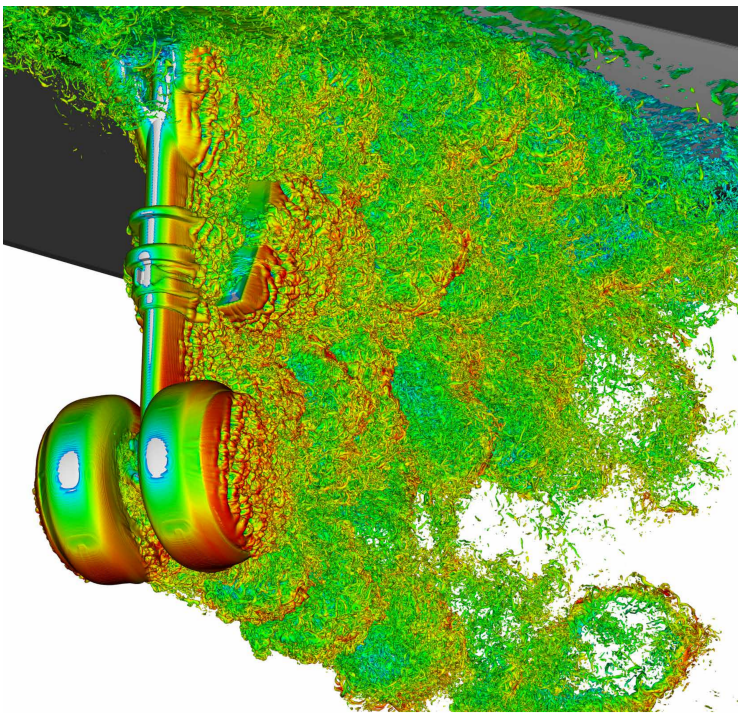


Figure 0.2: Simulation of the noise around a landing gear using the lattice Boltzmann method on a grid with twelve adapted grid levels. This results in a mesh with 2.28 billion cells and allows a reduction of the computation time by a factor of 15. The image shows the contours of the vorticity field and the colour code indicates the Mach number value. Courtesy of Michael Barad, Joseph Kocheemoolayil, NASA/Ames (Lattice Boltzmann for Airframe Noise Predictions).

during the convergence study. We observe that discrete velocity Boltzmann equations [Broadwell, 1964, Gatignol, 1975, Cabannes et al., 1980, Platkowski and Illner, 1988], while keeping the qualitative structure of the continuous velocity Boltzmann equation, notably by quadratic nonlinear collision terms, are not intended to convergence to the solution of this latter equation. This discussion will be further developed in Chapter 1.

The aim of this PhD thesis is to provide answers to two problems that affect lattice Boltzmann methods, described in the following.

1. The first challenge is to apply lattice Boltzmann methods to non-uniform computational grids. The objective of such a procedure is to reduce the cost of calculation and especially of storage by using fewer resources in places in space where a fine resolution is not necessary. As an example, a well-known scheme for dealing with three-dimensional problems, known as D_3Q_{27} (in what follows, a D_dQ_q scheme will be based on a d-dimensional lattice and will use q discrete velocities, following the notation in [Qian et al., 1992]), requires the storage of twenty-seven quantities at each point of the mesh. This can be prohibitive for realistic-size simulations with billions of cells, Figure 0.2, hence the interest in reducing the number of cells. It is therefore not a secondary concern but a necessity on the part of the industry, see NASA [Kiris et al., 2018] and CS Group (ProLB). However, this practical necessity should not obscure the associated issues from the point of view of the quality of the numerical solutions. Meeting these challenges requires a careful mathematical approach to control the error made, which is not possible with the heuristic approaches employed so far, and to minimise the spurious wave phenomena introduced by mesh adaptation.

2. The second difficulty concerns a mathematically rigorous understanding of lattice Boltzmann methods in their generality. In particular, a first point that remains to be clarified is the fact that these methods approach well the target PDEs, thus their consistency. A question that naturally follows concerns the study of the resilience of these schemes to perturbations, hence their stability. The combination of these two aspects can open the way to the study of the convergence of lattice Boltzmann schemes towards the solution of the target problem, making it possible to guarantee the good quality of the obtained numerical simulations.

The ambitious objective is therefore to bring lattice Boltzmann methods—in all their generality—into the disciplinary framework specific to numerical analysis, using the tools and concepts proper to this discipline. The goal is to have lattice Boltzmann methods eventually considered as numerical schemes for PDEs. This would foster the involvement of a community of researchers—who have powerful analysis tools but have hitherto been rather reluctant to tackle this type of problem—in a rediscovery of lattice Boltzmann schemes. This is all the more interesting as these methods are full of fundamental questions which remain open and which require new angles of attack. From the point of view of concrete applications of lattice Boltzmann methods, the interest is twofold. On the one hand, it will give new analysis tools leading to the design of schemes with better properties with respect to the physical model of interest. On the other hand, it will explain—from a fundamental point of view—a large number of good practices and know-how that have been developed "on the job" within the community of users of lattice Boltzmann methods.

Mesh adaptation for lattice Boltzmann methods

Referring the reader to the part of the manuscript dedicated to this first point for a more developed state of the art, we draw up here a general review of the existing studies, in order to identify general trends within the community. The two research directions we find are the following.

1. A first direction is to use a local time step at each grid level, which requires adapting the collision phase of the methods in order to preserve the physical parameters of the problem at the different grid resolutions [Filippova and Hänel, 1998, Lin and Lai, 2000, Kandhai et al., 2000, Yu et al., 2002, Crouse et al., 2003, Dupuis and Chopard, 2003, Rohde et al., 2006, Eitel-Amor et al., 2013, Feldhusen et al., 2016]. Most of the works following this approach use spatially adapted but time-fixed meshes. A paradigmatic study following this approach, which we analyse in detail, is that of [Filippova and Hänel, 1998]. The authors consider a BGK-type scheme with three conserved moments and a coarse mesh over the whole domain, on which patches at a finer level are superimposed, with an integer ratio n between space steps compared to the coarse mesh. Thus, the mesh is constructed according to prior knowledge of the problem. The relaxation parameter is adjusted between fine and coarse meshes in order to obtain the same viscosity coefficient at different resolutions. The numerical algorithm proceeds by first rescaling the solution only to the coarse meshes with superimposed fine cells. This involves the equilibrium values, the relaxation parameter and the n ratio. First, an iteration (transport and collision) is performed on the whole coarse mesh, which in this case brings the solution at time $t + \Delta t_{\text{coarse}}$. Second order space-time interpolations and a formula—again involving the values at equilibrium, the relaxation parameter, and the ratio n—allow us to construct the data at the edges of the fine patches at time $t, t + \Delta t_{\text{fine}}, \ldots, t + (n-1)\Delta t_{\text{fine}}$ in order to perform n iterations (transport and collision) on these fine cells. This completes one step of the algorithm. This approach has allowed the simulation of the *von Kármán* vortex street and to obtain values for the drag, lift and Strouhal number coefficients close to those of the reference. The only criticism raised in this work concerns the fact that, due to a dependence of the relaxation parameter in n, too many grid levels can make the method unstable.

2. A second trend is to use a global time step, regardless of the grid level, without the need to adapt the collision phase [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016]. Works in this direction rely on dynamically adapted meshes using refinement criteria specific to the problem under consideration. A representative work of this tendency is that of [Fakhari and Lee, 2014]. The authors tackle the MRT D_2Q_9 scheme with three conserved moments and consider a dynamically adapted mesh with different levels that communicate using ghost cells. The solution values on these cells are updated using interpolations. The refinement of the mesh is based on three criteria: a first one involving the vorticity; a second one based on the derivative of the vorticity; and finally a third one based on the ratio between the norm of the strain tensor and the rotation tensor. The time step is common to all grid levels, on which a collision step is made without any modification and a transport step employs Lax-Wendroff schemes, discretizing the transport equation associated with each discrete velocity. This approach has been validated on many test cases (Taylor-Green vortex, lid-driven flow, *von Kármán* vortex street, *etc.*) with results in agreement

with the reference values. Mesh adaptation allows for simulation time acceleration factors of up to four for some configurations. Compared to works in the spirit of [Filippova and Hänel, 1998], the authors of [Fakhari and Lee, 2014] point out that the absence of time interpolations—thanks to the choice of a global time step—allows one to avoid scaling the solution and to preserve the accuracy of the lattice Boltzmann method.

To summarize, with respect to the first problem addressed in this thesis, the available literature has introduced ways of adapting lattice Boltzmann schemes to non-uniform grids, using either meshes fixed at the beginning of the simulation [Filippova and Hänel, 1998, Lin and Lai, 2000, Kandhai et al., 2000, Dupuis and Chopard, 2003, Rohde et al., 2006], or dynamically adapted meshes with criteria taking into account the considered problem [Crouse et al., 2003, Eitel-Amor et al., 2013, Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016, Feldhusen et al., 2016]. This results in remarkable gains in terms of computation time and memory occupation. However, we observe that the strategies proposed so far are rarely time adaptive, which prevents the tracking of structures such as shock waves. Moreover, the available approaches do not allow to estimate and thus control the error committed by adapting the mesh. Moreover, they require—in the vast majority of cases—a profound modification of the underlying scheme as well as the definition of heuristic criteria for mesh adaptation, hence the lack of control over the error. Sometimes, even worse, they require the generation of a fixed mesh based on *a priori* knowledge of the flow being studied. Finally, grid adaptation is most often the cause of spurious waves at the interfaces between different resolutions [Gendre et al., 2017, Astoul et al., 2021], which pose difficulties in many applications, particularly in aeroacoustics.

These limitations justify the study that is the subject of Part I of this manuscript.

In Chapter 2, the approach consists first in proposing a dynamic mesh adaptation strategy based on adaptive multiresolution [Harten, 1994], which makes it possible to track various structures—such as shock waves, rarefaction waves, *etc.*—present in the solutions, and to control the error made by adapting the computational grid. Next, we propose a way to write general lattice Boltzmann schemes—always using multiresolution—so that they can be used on these adapted meshes by making "as if" we could utilize the scheme at the finest resolution, unlike the approach by [Fakhari and Lee, 2014], which feeds the numerical fluxes of the scheme at the finest level with information at the local grid level, Since the whole process is based on multiresolution, we prove that the error introduced by the methods on adapted grids compared to the reference one on a uniform mesh is controlled. This property is found in the numerous numerical experiments that follow, accompanied by a significant reduction in the memory trace of the lattice Boltzmann methods. This also shows the generality of our way of proceeding, beyond the peculiarities of the problem under consideration. The content of this chapter has been the subject of two publications: [Bellotti et al., 2022d] concerning the one-dimensional framework and [Bellotti et al., 2022c] for the multi-dimensional framework.

In Chapter 3, we study the proposed method in more depth. This is done by adapting the approach by equivalent equations [Dubois, 2008], which makes it possible to characterize the order from which our strategy perturbs the reference scheme. This shows that the behaviour of our method is at least one order closer to that of the reference scheme than the best approach available in the literature [Fakhari and Lee, 2014]. The crucial point allowing to obtain this high fidelity compared to existing approaches is not to solely compute the numerical flux at the finest resolution level with information coming from the local grid level, but to employ a reconstruction of the solution at the finest level in order to feed the fluxes. We also show that this has a significant impact on the amplitude of spurious waves at level jumps, which are substantially reduced using our approach. The content of this chapter makes up two publications: [Bellotti et al., 2022b, Bellotti et al., 2022a].

In Chapter 4, we adapt the analysis technique developed in the previous chapter in the context of lattice Boltzmann schemes in order to analyse the accuracy of adaptive Finite Volume methods based on multiresolution. This allows to quantify the order of perturbation as a function of the way numerical flows are computed. This information is then integrated into the standard error analysis for this type of approach and therefore provides

additional information on the behaviour of the schemes, in particular, with regard to smooth solutions where the mesh has been coarsened.

Beyond their theoretical design, we needed a computer implementation of the methods described in Part I. This is the subject of the work presented in Part II.

In Chapter 5, we present the salient features of the C++ library SAMURAI, which allows us to tackle the question of mesh adaptation from a general point of view, without restricting ourselves to the volume-based multiresolution which is the subject of this thesis. To do this, we synthesise any non-uniform Cartesian mesh by grouping the cells level-by-level and according to their spatial connectivity, in order to represent this by intervals of integers. This results in a compressed encoding of the mesh which allows the introduction of algebraic operations on sets. This is used to select sub-parts of the mesh in order to perform operations on them in an easy and transparent way. We make a judicious choice concerning the numbering of cells for storage purposes, which allows optimized memory access. Moreover, the employed storage relies on the `xtensor` library, in order to write mathematical expressions on the stored fields associated with the mesh in all simplicity and to benefit from the lazy evaluation of these expressions.

In Chapter 6, we specialise SAMURAI in adapting the mesh *via* multiresolution and we implement the corresponding adaptive lattice Boltzmann method. This is not done by a recurrent but rather using an iterative implementation, based not on trees of cells but on different categories of cells. The implementation of the numerical methods conceived during the thesis has been the subject of a work which extends over three years, in continuous interaction with expert developers, in particular Loïc Gouarin. This is part of the HPC@Maths project and has provided a growing number of researchers around the project with an environment that meets their needs for mesh adaptation and multiresolution. Another medium-term objective pursued within the project is to integrate the mesh adaptation of SAMURAI into the software package pyLBM, which allows general lattice Boltzmann schemes to be simply represented and implemented with parallelization through automatic code generation.

Consistency, stability and convergence of lattice Boltzmann methods

Concerning the second major issue addressed in this thesis, we mention that numerous consistency and stability analysis procedures based on formal arguments exist. Their formal character comes from the fact that they give results in agreement with numerical simulations without focusing on mathematical rigour. As far as consistency is concerned, we note the "historical" approach called Chapman-Enskog expansion [Chen and Doolen, 1998, Qian and Zhou, 2000]; the equivalent equations [Dubois, 2008, Dubois, 2022]; Maxwell iteration [Yong et al., 2016, Zhao and Yong, 2017]; as well as the asymptotic analyses of [Junk and Yong, 2003, Junk et al., 2005, Junk and Yang, 2009]. Concerning the stability analysis, the most classical approach consists in carrying out a *von Neumann* analysis on the linear/linearized schemes [Benzi et al., 1992, Sterling and Chen, 1996, Lallemand and Luo, 2000, Graille, 2014, Février, 2014], verifying that the eigenvalues of the matrix of the scheme—written using the discrete Fourier transform—are inside the unit circle uniformly in the wave-number. These questions will be the subject of a specific state of the art in Part III. Beyond these works, we now try to identify two main movements within the research community aiming at analyses of the consistency and stability of lattice Boltzmann schemes with the tools proper to numerical analysis.

1. A first angle of attack is based on the original formulation of lattice Boltzmann methods, in which the conserved moments (those of interest in the studied PDEs) and the non-conserved moments (of purely numerical nature) are all present in the numerical scheme [Junk and Yang, 2009, Junk and Yang, 2015, Caetano et al., 2023]. The work of [Caetano et al., 2023] is a typical example of this trend. The authors study the convergence of the D_1Q_2 scheme towards the entropic weak solution of a scalar conservation law. The study is conducted in the under-relaxation regime with initial data at equilibrium. The first key step is to prove a maximum principle for the conserved moment and velocity distributions. This provides estimates of the total variation in time and space, as well as quantifications of the deviation of the non-conservative moment from its value at equilibrium. *In fine*, by combining these arguments, the convergence of the numerical solution to a weak—non-unique—solution of the target PDE is proved. Relying on kinetic entropies, the

authors construct numerical entropy-entropy flux pairs and the associated entropy inequalities. Particular attention is paid to the fact that these entropies are evaluated on quantities defined after the collision, just before the stream step. This makes it possible to prove the convergence of the numerical solution to the unique entropy solution of the PDE for the L^1 norm. Two limitations, clearly identified by the authors, are the fact of not being able to study the over-relaxation regime as well as approaching the study of schemes with more discrete velocities.

2. A second angle of attack proceeds first to the rewriting of the original lattice Boltzmann scheme under the form of a "corresponding" scheme, in which only the conserved moments remain present [Ginzburg, 2009, Suga, 2010, Kuzmin et al., 2011, Lin et al., 2021, Dellacherie, 2014, Fučík and Straka, 2021]. The work of [Dellacherie, 2014] is a good example of this way of proceeding. The author of this publication considers a linear D_1Q_2 scheme. The central point is an algebraic calculation, carried out by writing the original numerical scheme on several time steps at different points in space, which allows, by recombining these expressions, to eliminate the non-conservative moment. This results in a rewriting of the scheme in the form of a multi-step Finite Difference method on the single conserved moment. Consequently, this rigorously shows that the scheme is consistent with the target PDE and we can easily determine the stability conditions for the L^2 and L^∞ norms. Finally, the Lax theorem [Lax and Richtmyer, 1956] allows to conclude on the convergence of the scheme—provided that it is stable—to the regular solutions of the PDE in the chosen norm. An important open question is identified here: how to use the same approach to study more complicated lattice Boltzmann schemes or that approximate the solutions of nonlinear PDEs.

To summarize, previous works have shown that some simple lattice Boltzmann schemes (D_1Q_2 [Dellacherie, 2014], D_1Q_3 [Suga, 2010, Lin et al., 2021] and D_dQ_{2W+1} TRT [Ginzburg, 2009, Kuzmin et al., 2011], typically with only one conserved moment) can be re-written under the form of Finite Difference schemes on the conserved moment, which has shed light on the correct notions of consistency, stability, and convergence of these methods. Other works [Junk and Yang, 2009, Junk and Yang, 2015, Caetano et al., 2023] act directly on the original scheme, sometimes by comparing it to relaxation schemes [Jin and Xin, 1995, Aregba-Driollet and Natalini, 2000]. However, writing in Finite Difference form or interpreting schemes as a relaxation scheme only apply to very simple schemes and it is not clear how this can be generalized to any lattice Boltzmann method. Finally, the methods of formal analysis mentioned at the beginning apply to a wide range of schemes with significant predictive power but have not been integrated, as yet, into a programme that could be summarised as "consistency and stability imply convergence".

Beyond the questions of consistency and stability of lattice Boltzmann methods, it is important to take the role of the initialization of the schemes as well as the presence of a bounded computational domain, which therefore requires boundary conditions to be enforced, into account. As for the first point, the initialization of the lattice Boltzmann schemes can be done in a partially arbitrary way because these schemes involve more computational variables than initial data in the problem at hand. Even though the most common way is to initialise the missing variables at equilibrium, various works [Caiazzo, 2005, Van Leemput et al., 2009, Huang et al., 2015b] have proposed formal analyses aimed at obtaining a finer understanding of the behaviour of the schemes close to the initial time as a function of the choice of initialisation, which is not necessarily at equilibrium. These issues are very important from a practical point of view, in particular in order not to reduce the order of the methods due to bad initializations. Concerning the boundary conditions, the practical implementation of the lattice Boltzmann methods requires carrying out the calculations on a bounded grid. This leads to a lack of information during the stream phase at the edges of the domain, which must be filled by imposing a boundary condition, for the moment of a purely numerical nature. In addition, these numerical boundary conditions on the distribution functions can be used to impose physical boundary conditions (inflow, no-slip, *etc.*) specific to the system of PDEs to approximate, expressed only on the conserved moments of the problem. This discrepancy between purely numerical and physical boundary conditions remains a major difficulty within the community. Moreover, the boundary conditions can introduce phenomena of numerical instability that must be understood and mastered. There is a very large body of work on boundary conditions for specific problems and schemes, see [Krüger et al., 2017, Chapter 5]. We will not detail here all these contributions. We just mention two of the most widespread approaches. The first is called the "bounce-back" condition [Ginzbourg and Adler, 1994, Bouzidi et al., 2001, Dubois et al., 2015] and con-

sists in filling the missing information concerning the distribution of a velocity coming from outside the domain with that of the opposite velocity. This allows—for example—to impose no-slip conditions on the velocity field for D_2Q_9 schemes with three conserved moments. A second condition very much in use is called "anti-bounce-back" [Ginzburg, 2005, Ginzburg et al., 2008a, Dubois et al., 2020b]. The missing information about the distribution for a velocity coming from outside the domain is replaced by the opposite value of that of the opposite velocity. This makes it possible to impose—if used on a D_2Q_9 scheme with three conserved moments—a Dirichlet condition on the pressure field, thus on the density. Generally speaking, we lack tools for analysing boundary conditions and this has important practical repercussions, firstly in terms of the fidelity of the numerical conditions to the physical conditions that we would like to impose and secondly on the practical possibility of using certain conditions, because of their instability.

These open questions justify our study in Part III and Part IV.

In Chapter 7, the approach consists in eliminating the non-conserved moments within any lattice Boltzmann method using the algebraic properties of a commutative ring. Indeed, these moments are not present in the target PDEs. This is possible because the Cayley-Hamilton theorem applies to matrices whose elements belong to a commutative ring. Once the non-conservative moments have been eliminated, we are left with multi-step Finite Difference schemes on the conserved variables, for which the notions of consistency, stability, and convergence are classical and a large number of results are already available in the literature. The content of this chapter is included in the article [Bellotti et al., 2022e].

In Chapter 8, we study the consistency of lattice Boltzmann methods using the "corresponding" Finite Difference schemes. However, this is done without explicitly writing the corresponding method for the considered lattice Boltzmann scheme down, but by characterizing in a sufficiently precise way the transformation from the original scheme to the scheme without the non-conserved moments. This is made possible by the Taylor expansions of the determinant and adjugate (also known as classical adjoint) matrix of a given matrix. Thus, we find the modified equations for general schemes and we also corroborate the results coming from two formal techniques of analysis [Dubois, 2022, Yong et al., 2016]. The material of this chapter is presented in [Bellotti, 2023b].

In Chapter 9, we relate the notion of stability *à la von Neumann* for multi-step Finite Difference schemes with that for lattice Boltzmann schemes. The latter is indeed the most widespread within the community of lattice Boltzmann methods. This link is made possible—again—by having characterized the transformation of the lattice Boltzmann scheme to the Finite Difference scheme with precision. This gives a rigorous character to a procedure that was previously used in an "intuitive" way.

In Chapter 10, we study the question of the initialization of lattice Boltzmann scheme, which has important effects on the order of convergence as well as on the formation of time boundary layers on the numerical solution. Indeed, this problem arises because the lattice Boltzmann methods in their original formulation have more variables to initialize than those present in the PDEs to approximate, while once written as Finite Difference schemes, these schemes are multi-step and therefore require initialization routines. To this end, we propose an analysis based on the modified equations of the schemes close to the initial time, allowing to secure initializations preserving the order of the methods and to avoid oscillations at the beginning of the simulations. Moreover, by introducing the notion of "observability" of a lattice Boltzmann scheme, we identify a class of schemes for which the initialization is easy to study with the tools introduced in this work. The content of this chapter is the subject of a preprint [Bellotti, 2023a] submitted to a peer-reviewed journal.

In Chapter 11, which constitutes preliminary work concerning the nonlinear stability and monotonicity of lattice Boltzmann methods using the corresponding Finite Difference scheme, we demonstrate the convergence of the D_1Q_2 lattice Boltzmann scheme to the weak entropic solution of a scalar conservation law in the over-relaxation regime with initial data at equilibrium. The study is possible by working on the "corresponding" Finite Difference scheme, trivially generalizing the notion of monotone scheme to multi-step methods. We thus demonstrate a maximum principle on the conserved moment, estimates on the total variation in space and time on the conserved moment. By working with Krushkov entropy-entropy flux pairs, we establish a multi-step discrete entropy

inequality, which allows us to conclude. The unfinished part of this work would be to study the under-relaxation regime *via* the Finite Difference scheme in order to recover the results by [Caetano et al., 2023]. Some points in the proof have been established but others remain open. It can be said that the properties in the under-relaxation regime heavily rely on the initialization at equilibrium, which must be correctly taken into account, and question the universality of a trivial generalization of monotonicity to multi-step methods [Hundsdorfer et al., 2003].

Finally, in Chapter 12, we start a preliminary work concerning the consistency and stability of the boundary conditions for lattice Boltzmann methods. In particular, we focus on two one-dimensional schemes—namely D_1Q_2 and D_1Q_3—which can be written, even at the boundary for the boundary conditions at hand, as Finite Difference schemes. This allows an analysis of consistency by Taylor expansions and of stability by following [Gustafsson et al., 1972]. We also propose a formal procedure for consistency analysis based on Maxwell iteration [Yong et al., 2016], which gives the same results as the passage through the Finite Difference scheme. However, the approach *via* the elimination of conserved moments remains highly limited—when boundary conditions are imposed—because in this case we lack a general algebraic result playing the same role as the Cayley-Hamilton theorem in the case of an unbounded domain/periodic conditions. This can be seen, even on the simplest schemes, from the difficulties in eliminating non-conserved moments for general boundary conditions.

Publications, presentations and collaborations

This PhD thesis resulted in the following scientific contributions.

Articles in peer-reviewed journals

- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022d). Multiresolution-based mesh adaptation and error control for lattice Boltzmann methods with applications to hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 44(4):A2599–A2627
- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022c). Multidimensional fully adaptive lattice Boltzmann methods with error control based on multiresolution analysis. *Journal of Computational Physics*, 471:111670
- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022b). High accuracy analysis of adaptive multiresolution-based lattice Boltzmann schemes via the equivalent equations. *SMAI Journal of Computational Mathematics*, 8:161–199
- Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022a). Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps? *Comptes Rendus. Mathématique*, 360:761–769
- Bellotti, T., Graille, B., and Massot, M. (2022e). Finite difference formulation of any lattice Boltzmann scheme. *Numerische Mathematik*, 152:1–40
- Bellotti, T. (2023b). Truncation errors and modified equations for the lattice Boltzmann method via the corresponding Finite Difference schemes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 57(3):1225–1255

Preprints and in-preparation articles

- Bellotti, T. (2023a). Initialisation from lattice Boltzmann to multi-step Finite Difference methods: modified equations and discrete observability. *Submitted, see https://hal.science/hal-03989355*
- Bellotti, T., Gouarin, L., Leclerc, H., Massot, M., and Séries, L. (2023a). Interval-based data structure for Cartesian meshes: application to multi-scale PDEs on adaptive meshes. *In Preparation*
- Bellotti, T., Massot, J., Massot, M., Séries, L., and Tenaud, C. (2023b). Modified equation and error analyses of adaptive multiresolution Finite Volume schemes. *In Preparation*

Talks

- Initialization from lattice Boltzmann to multi-step Finite Difference methods: modified equations and discrete observability, *invitation at 19th International Conference for Mesoscopic Methods in Engineering and Science*, Chengdu (China), July 2023.
- Initialization from lattice Boltzmann to multi-step Finite Difference methods: modified equations and discrete observability, *22th IACM - CFC 2023*, Cannes (France), April 2023.
- Finite Difference formulation of any lattice Boltzmann scheme: consistency and stability (and initialization), *invitation at the Groupe de Travail d'EDP et Calcul Scientifique di LMRS et LMI*, Rouen (France), February 2023.
- Finite Difference formulation of any lattice Boltzmann scheme: consistency and stability, *invitation at the séminaire de l'IRMAR*, Rennes (France), December 2022.
- Initialisation pour les schémas de Boltzmann sur réseau : aspects discrets et asymptotiques, *invitation at the workshop "Schémas numériques de type Boltzmann"*, Bordeaux (France), November 2022.
- Finite Difference formulation of any lattice Boltzmann scheme: consistency and stability, *invitation at the séminaire EDP*, Strasbourg (France), October 2022.
- Lattice Boltzmann schemes on dynamically adapted meshes relying on multiresolution and recent theoretical breakthroughs in the understanding of lattice Boltzmann methods, *talk at von Karman Institute*, Bruxelles (Belgium), September 2022.
- Limits of the Kinetic Interpretation of Lattice Boltzmann Schemes: A Cure via a Macroscopic Standpoint with Finite Difference Schemes on the Conserved Moments, *32nd Internation Symposium on Rarefied Gas Dynamics*, Seoul (South Korea), July 2022.
- Finite Difference formulation of lattice Boltzmann schemes: consequences on consistency and stability, *18th International Conference for Mesoscopic Methods in Engineering and Science*, La Rochelle (France), June 2022.
- Consistency and stability of lattice Boltzmann schemes, *45ème Congrès d'Analyse Numérique (CANUM)*, Évian-les-Bains (France), June 2022.
- Finite Difference formulation of any lattice Boltzmann scheme, *invitation at the séminaire Analyse Numérique et EDP*, Orsay (France), March 2022.
- Finite Difference formulation of any lattice Boltzmann scheme, *invitation at the 34ème séminaire CEA/GAMNI*, Paris (France), January 2022.
- Une formulation de type différences finies pour les scheémas de Boltzmann sur reéseau, *Groupe de travail "Schémas de Boltzmann sur réseau"*, Paris (France), November 2021.
- Une formulation de type différences finies pour les scheémas de Boltzmann sur reéseau, *invitation au workshop " Modèles et méthodes pour les équations cinétiques"*, Bordeaux (France), October 2021.
- Fully adaptive lattice Boltzmann methods based on multiresolution analysis: error control, high accuracy, equivalent equations and grid jumps, *17th International Conference for Mesoscopic Methods in Engineering and Science*, online, July 2021.
- Fully adaptive lattice Boltzmann methods with error control based on multiresolution analysis, *10ème Biennale Française des Mathématiques Appliquées*, La Grande-Motte (France), June 2021.
- Adaptive multiresolution-based lattice Boltzmann schemes and their accuracy analysis *via* the equivalent equations, *Groupe de travail "Schémas de Boltzmann sur réseau"*, Paris (France), May 2021.
- A class of multidimensional fully adaptive lattice-Boltzmann methods based on multiresolution analysis, *WCCM ECCOMAS 2020*, online, January 2021.

Collaborations et interactions

During the three years of my thesis, I had the opportunity to interact and collaborate, beyond my direct supervisors (Benjamin Graille, Loïc Gouarin and Marc Massot), with the following researchers.

- Laurent Séries, concerning the development of SAMURAI.
- Christian Tenaud, about multiresolution and its application to Finite Volume schemes.
- Louis Reboul, to consider a possible link between lattice Boltzmann schemes and “asymptotic preserving” methods.
- Thierry Magin, about the use of lattice Boltzmann methods to tackle multi-phase flows.

Chapter 1

Lattice Boltzmann methods

The aim of Chapter 1 is to provide a brief historical introduction as well as the fundamental notations concerning lattice Boltzmann schemes. These shall be used throughout the entire manuscript.

1.1 Brief historical introduction

Lattice gas automata are the ancestors of the lattice Boltzmann schemes and have been used since the seventies to study fluid flows. The automaton proposed by [Hardy et al., 1973] is based on a two-dimensional Cartesian lattice where each site can be occupied by four populations of particles with corresponding velocity along the Cartesian axes. At each node of the lattice, the presence/absence of a particle is represented by a binary variable. At each iteration in the "life" of the automaton, particles move according to their velocity and collision rules are proposed in the case where two particles collide head-on. Another model, based on hexagonal lattices, has been later proposed by [Frisch et al., 1986] and features non-deterministic collision rules. An important drawback of these automata in the simulation of equations such as the Navier-Stokes system is that macroscopic quantities are retrieved by spatial average and are thus subjected to a strong statistical noise. Moreover, these models introduce non-Galilean terms in the Navier-Stokes equations.

The birth of the actual lattice Boltzmann method [McNamara and Zanetti, 1988, Higuera and Jiménez, 1989, Higuera et al., 1989] dates back to the substitution of Boolean variables—used to indicate the presence/absence of a particle with given discrete velocity at a site of the lattice—with distribution densities. These distribution densities are interpreted—in a manner germane to statistical physics—as probability densities relative to the presence of particles at a given discrete velocities in the neighborhood of a site. At each time step of the method, particles undergo a collision phase which is encoded in terms of distribution functions, followed by a free stream phase according to their velocity. The collision operators are always linearized around equilibria and first feature a unique relaxation time (BGK approximation) [Qian et al., 1992]. However, the introduction of multiple relaxation times (MRT) has been rapidly proposed [D'Humières, 1992]. A simple way of dealing with several relaxation parameters is to introduce a change of basis on the distribution functions, yielding the so-called "moments", and rewrite the collision as a diagonal relaxation of the moments, each one with its relaxation time, towards their respective equilibrium.

In this work, lattice Boltzmann methods (sometimes abridged by "LBM") are seen as a class of numerical schemes used to solve numerous problems in applied mathematics and fluid mechanics, coming under the form of $N \in \mathbb{N}^*$ conservation laws. These methods are applied to uniform Cartesian time-space lattices. As any numerical method for time-space equations, they rely on a particular link between the temporal and the spatial discretizations. Moreover, they stem from a finite family of discrete velocities, a local collision phase, and a linear stream phase. We introduce lattice Boltzmann methods phenomenologically as given numerical algorithms. The schemes that we consider in our work fall within the class of MRT schemes [D'Humières, 1992].

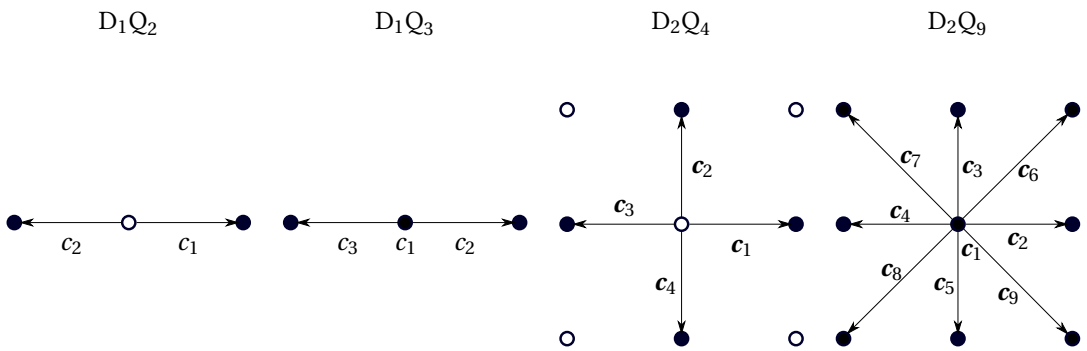


Figure 1.1: Examples of common sets of discrete velocities.

1.2 Time and space discretization

We consider the problem to be set on $\mathbb{R}_+ \times \mathbb{R}^d$ where the dimension of the space is $d = 1, 2, 3$. For the moment, we do not deal with any boundary condition.

- Time is uniformly discretized with step $\Delta t > 0$, thus considering a lattice $\Delta t \mathbb{N}$.
- Space is uniformly discretized with step $\Delta x > 0$, thus considering a lattice $\Delta x \mathbb{Z}^d$.

The so-called "lattice velocity" $\lambda > 0$ is defined by

$$\lambda := \frac{\Delta x}{\Delta t},$$

and has the dimension of a velocity. It represents the speed of propagation of information on the discrete lattice. When the lattices are refined to analyze the convergence regime, time Δt and space Δx steps need to be linked through a specific scaling $\Delta t = \Lambda(\Delta x)$ where $\lim_{\xi \to 0} \Lambda(\xi) = 0$ and $\Lambda > 0$. Literature features two main scalings, which are

- the "acoustic" scaling [Dubois, 2008, Graille, 2014, Caetano et al., 2023, Dubois, 2022], where $\Lambda(\xi) \propto \xi$, hence λ remains fixed as $\Delta x \to 0$. This means that even when reducing Δx, the speed of propagation of information—*i.e.* λ—remains finite.
- the "diffusive" scaling [Junk and Yang, 2015, Zhao and Yong, 2017, Zhang et al., 2019], where $\Lambda(\xi) \propto \xi^2$, hence $\lambda \propto \Delta x^{-1}$ as $\Delta x \to 0$. This means that when reducing Δx, the speed of propagation of information—*i.e.* λ—becomes infinite.

The behavior of the scheme and the approximated equations are different according to the choice of scaling Λ, which is not harmless as for Finite Difference schemes, see [Allaire, 2007, Remark 2.3.3]. Even if the acoustic scaling shall be used in most of the manuscript, we sometimes consider the diffusive scaling. This shall be indicated explicitly.

1.3 Discrete velocities and particle distribution functions

As previously claimed, one important choice to be made when proposing a lattice Boltzmann scheme is the choice of the set of $q \in \mathbb{N}^*$ discrete velocities. We shall indicate them by $(\boldsymbol{\xi}_j)_{j \in [\![1,q]\!]} \subset \mathbb{R}^d$. These discrete velocities are integer multiple of the lattice velocity λ, thus for every $j \in [\![1,q]\!]$, there exists $\boldsymbol{c}_j \in \mathbb{Z}^d$ such that $\boldsymbol{\xi}_j = \lambda \boldsymbol{c}_j$. This guarantees—as we shall see—that the "virtual particles" are stuck to dwell on the lattice $\Delta x \mathbb{Z}^d$ at each time-step of the method. We observe that the family of discrete velocities is generally taken to be symmetric, in the sense that for each discrete velocity, its opposite is also considered. This is dictated by isotropy arguments and is beneficial when imposing boundary conditions but is not strictly needed. We denote the distribution density of particles moving at velocity $\boldsymbol{\xi}_j$ by $\mathsf{f}_j = \mathsf{f}_j(t, \boldsymbol{x})$ for $(t, \boldsymbol{x}) \in \Delta t \mathbb{N} \times \Delta x \mathbb{Z}^d$ for every $j \in [\![1,q]\!]$. It is customary to call the

schemes $\mathrm{D}_d\mathrm{Q}_q$ because they are set on a d-dimensional lattice and because they employ q discrete velocities. Figure 1.1 provides examples of sets of discrete velocities that we will consider.

1.4 Numerical algorithm: collide and stream

As previously mentioned, any lattice Boltzmann scheme consists in a kinetic algorithm made up of two phases: a local collision phase performed on each site of the lattice $\Delta x\mathbb{Z}^d$ and a stream phase where particles are exchanged between different sites of the lattice. The solution undergoes these two phases to evolve from $t \in \Delta t\mathbb{N}$ to $t + \Delta t \in \Delta t\mathbb{N}$.

1.4.1 Collision

The collision phase is a linear relaxation local to each site of the lattice. In the formalism by [D'Humières, 1992], instead of writing it on $\mathbf{f} = (\mathsf{f}_1, \dots, \mathsf{f}_q)^{\mathsf{t}}$, this phase is diagonalized using a change of basis called moment matrix $\boldsymbol{M} \in \mathrm{GL}_q(\mathbb{R})$. In this way, the collision is written on the moments $\mathbf{m}$ obtained by $\mathbf{m} = \boldsymbol{M}\mathbf{f}$. It reads

$$\mathbf{m}^\star(t, \boldsymbol{x}) = (\boldsymbol{I} - \boldsymbol{S})\mathbf{m}(t, \boldsymbol{x}) + \boldsymbol{S}\boldsymbol{m}^{\mathrm{eq}}(\mathsf{m}_1(t, \boldsymbol{x}), \dots, \mathsf{m}_N(t, \boldsymbol{x})), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d. \tag{1.1}$$

We shall soon see why the first N moments, forming the variables of interest, are called "conserved moments". We list them before the non-conserved ones for the sake of notation. In what follows, the $\star$ indicates any post-collision state (*i.e.* $\mathbf{f}^\star = \boldsymbol{M}^{-1}\mathbf{m}^\star$ at each grid point). Let us introduce each term in (1.1).

- The matrix $\boldsymbol{I} \in \mathrm{GL}_q(\mathbb{R})$ is the identity matrix of size q.
- The matrix $\boldsymbol{S} \in \mathcal{M}_q(\mathbb{R})$ is the "relaxation matrix" containing the relaxation parameters, under the form

  $$\boldsymbol{S} = \mathrm{diag}(s_1, \dots, s_N, s_{N+1}, \dots, s_q), \tag{1.2}$$

  where the relaxation parameters for the conserved moments $s_i \in \mathbb{R}$ for $i \in [\![1, N]\!]$ and the ones relative to the non-conserved moments $s_i \in]0, 2]$ for $i \in [\![N+1, q]\!]$ are taken between zero and two [Higuera et al., 1989, Benzi et al., 1992, Dubois, 2008] for reasons linked with stability.
- The moments at equilibrium $\boldsymbol{m}^{\mathrm{eq}} : \mathbb{R}^N \to \mathbb{R}^q$ are possibly non-linear functions of the N conserved moments. In order to guarantee that the first N moments are conserved through the collision phase, irrespective of the values of $s_1, \dots, s_N$, the constraints

  $$m_i^{\mathrm{eq}}(\mathsf{m}_1, \dots, \mathsf{m}_N) = \mathsf{m}_i, \qquad \forall i \in [\![1, N]\!], \tag{1.3}$$

  must hold [Bouchut et al., 2000], hence guaranteeing $\mathsf{m}_i^\star = \mathsf{m}_i$ at any lattice point. Although it is often customary [Février, 2014, Chapter 1] to consider $s_1, \dots, s_N = 0$ to show which moments are conserved at a glance, having $\mathrm{rank}(\boldsymbol{S}) = q - N$, this is not compulsory and shall sometimes be avoided for the sake of simplifying the discussion, *cf.* Chapter 8.

1.4.2 Stream

Once the collision phase has been done, it is time to perform the stream/transport phase according to the choice of discrete velocities at hand. This phase is non-local but linear and corresponds to a shift of the data along the characteristics of each velocity field. Since the discrete velocities are multiple of the lattice velocity λ, any consistent discretization of the transport equation at these velocities ends up to be exact and given by the upwind formulæ

$$\mathsf{f}_j(t + \Delta t, \boldsymbol{x}) = \mathsf{f}_j^\star(t, \boldsymbol{x} - \boldsymbol{\xi}_j \Delta t) = \mathsf{f}_j^\star(t, \boldsymbol{x} - \boldsymbol{c}_j \Delta x), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d, \tag{1.4}$$

for $j \in [\![1, q]\!]$. Observe that information needed by (1.4) belongs to the lattice $\Delta x\mathbb{Z}^d$ thanks to the fact that the discrete velocities are multiple of the lattice velocity λ.

1.5 Examples

To make everything more concrete, we present some examples of lattice Boltzmann schemes that will be used in the sequel.

1.5.1 D_1Q_2

The so-called D_1Q_2 [Dellacherie, 2014, Graille, 2014] is probably the simplest lattice Boltzmann scheme. It is obtained for $d = 1$ taking two discrete velocities, thus $q = 2$ and $c_1 = 1$ and $c_2 = -1$. The usual moment matrix is

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 \\ \lambda & -\lambda \end{bmatrix}, \qquad \text{or its dimensionless version} \qquad \boldsymbol{M} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Usually, the dimensional matrix $\boldsymbol{M}$ is considered, especially under acoustic scaling. In the sequel of the work, when we want to write the scheme to compare its behavior both under acoustic and diffusive scaling, we shall use the dimensionless version. The reason is that under diffusive scaling, the entries of the dimensional matrix $\boldsymbol{M}$ have different trends with respect to Δx, in particular $M_{11}, M_{12} = O(1)$ and $M_{21}, M_{22} = O(1/\Delta x)$, whereas we would like to have any entry fixed as $\Delta x \to 0$.

Remark 1.5.1 (Dimensionless matrices). *From now on, when we talk about dimensionless moment matrix, this corresponds to take the dimensional moment matrix (where terms in λ) and set all $\lambda = 1$.*

Due to the limited number of degrees of freedom, the D_1Q_2 scheme can be employed only with one conserved moment $N = 1$. This allows to use this scheme, upon choosing the equilibrium m_2^{eq} correctly, to approximate the solution of a scalar conservation law (advection equation, Burgers equation, *etc.*) [Graille, 2014] under acoustic scaling.

1.5.2 D_1Q_3

This scheme [Février, 2014, Dubois et al., 2020a] features $d = 1$ taking one additional zero velocity compared to the D_1Q_2 scheme. We consider three discrete velocities $q = 3$ and $c_1 = 0$, $c_2 = 1$ and $c_3 = -1$. Several choices for the moment matrix $\boldsymbol{M}$ can be considered. The first one is is

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & \lambda & -\lambda \\ 0 & \lambda^2 & \lambda^2 \end{bmatrix}. \tag{1.5}$$

Another possible moment matrix is

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & \lambda & -\lambda \\ -2\lambda^2 & \lambda^2 & \lambda^2 \end{bmatrix}, \tag{1.6}$$

where rows are orthogonal with respect to the Euclidian scalar product of $\mathbb{R}^3$. Now that the scheme has a larger number of degrees of freedom compared to the D_1Q_2, it can be used with one $N = 1$ conserved moment to approximate—under acoustic scaling—the solution of a scalar conservation law or with two $N = 2$ conserved moments to handle—again under acoustic scaling—the solution of a wave equation or the shallow-water system.

1.5.3 D_2Q_4

Considering a two-dimensional setting, the D_2Q_4 [Mohamad and Kuzmin, 2012, Février, 2014] is the simplest scheme with symmetric discrete velocities. It is found considering $q = 4$ and the discrete velocities

$$\boldsymbol{c}_1 = (1,0)^{\mathrm{t}}, \qquad \boldsymbol{c}_2 = (0,1)^{\mathrm{t}}, \qquad \boldsymbol{c}_3 = (-1,0)^{\mathrm{t}}, \qquad \boldsymbol{c}_4 = (0,-1)^{\mathrm{t}},$$

along with the moment matrix

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ \lambda & 0 & -\lambda & 0 \\ 0 & \lambda & 0 & -\lambda \\ \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 \end{bmatrix}.$$

The scheme is usually employed with $N = 1$ conserved moment to approximate—under acoustic scaling—the solution of a two-dimensional scalar conservation law. Another possibility [Février, 2014, Chapter 1] is to consider discrete velocities which are not parallel to the axes but rather along the diagonals

$$\boldsymbol{c}_1 = (1,1)^{\mathrm{t}}, \qquad \boldsymbol{c}_2 = (-1,1)^{\mathrm{t}}, \qquad \boldsymbol{c}_3 = (-1,-1)^{\mathrm{t}}, \qquad \boldsymbol{c}_4 = (1,-1)^{\mathrm{t}},$$

yielding the so-called "twisted" scheme.

1.5.4 $\mathrm{D_2Q_9}$

This is probably the most well-known two-dimensional $d = 2$ lattice Boltzmann scheme, at the point that sometimes the term "lattice Boltzmann scheme" is taken as a synonym of $\mathrm{D_2Q_9}$. The scheme [Qian et al., 1992] comes with $q = 9$ and discrete velocities

$$\begin{aligned} &\boldsymbol{c}_1 = (0,0)^{\mathrm{t}}, \\ &\boldsymbol{c}_2 = (1,0)^{\mathrm{t}}, \qquad \boldsymbol{c}_3 = (0,1)^{\mathrm{t}}, \qquad \boldsymbol{c}_4 = (-1,0)^{\mathrm{t}}, \qquad \boldsymbol{c}_5 = (0,-1)^{\mathrm{t}}, \\ &\boldsymbol{c}_6 = (1,1)^{\mathrm{t}}, \qquad \boldsymbol{c}_7 = (-1,1)^{\mathrm{t}}, \qquad \boldsymbol{c}_8 = (-1,-1)^{\mathrm{t}}, \qquad \boldsymbol{c}_9 = (1,-1)^{\mathrm{t}}. \end{aligned}$$

A vast variety [Février, 2014] of moment matrices $\boldsymbol{M}$ exists. We are not going to discuss the role of the different choices for $\boldsymbol{M}$ and we shall stick with the one proposed by [Lallemand and Luo, 2000]:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & \lambda & 0 & -\lambda & 0 & \lambda & -\lambda & -\lambda & \lambda \\ 0 & 0 & \lambda & 0 & -\lambda & \lambda & \lambda & -\lambda & -\lambda \\ -4\lambda^2 & -\lambda^2 & -\lambda^2 & -\lambda^2 & -\lambda^2 & 2\lambda^2 & 2\lambda^2 & 2\lambda^2 & 2\lambda^2 \\ 0 & -2\lambda^3 & 0 & 2\lambda^3 & 0 & \lambda^3 & -\lambda^3 & -\lambda^3 & \lambda^3 \\ 0 & 0 & -2\lambda^3 & 0 & 2\lambda^3 & \lambda^3 & \lambda^3 & -\lambda^3 & -\lambda^3 \\ 4\lambda^4 & -2\lambda^4 & -2\lambda^4 & -2\lambda^4 & -2\lambda^4 & \lambda^4 & \lambda^4 & \lambda^4 & \lambda^4 \\ 0 & \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 \end{bmatrix},$$

This scheme can be used with $N = 1$ and *ad hoc* equilibria to simulate the advection-diffusion equation [Zhang et al., 2019] and with $N = 3$ to simulate the incompressible Navier-Stokes equations [Lallemand and Luo, 2000, Geier et al., 2006].

1.6 Parameters tuning in the lattice Boltzmann schemes

We have presented lattice Boltzmann as a sort of "recipe" with a certain number of "ingredients", without detailing how to select them according to the problem to solve. In particular, the choice of scaling Λ, discrete velocities $(\boldsymbol{c}_j)_{j\in[\![1,q]\!]}$, moment matrix $\boldsymbol{M}$, relaxation parameters $s_{N+1},\dots,s_{q+1}$ and equilibria $\boldsymbol{m}^{\mathrm{eq}}$ influences what are the PDEs which solution is approximated by the lattice Boltzmann scheme at hand. This topic is covered by the consistency analyses proposed for the lattice Boltzmann schemes, see Chapter 8. Moreover, these choices—for numerous free leverages are present—influence the stability of the lattice Boltzmann methods, *cf.* Chapter 9 and Chapter 11, and must be tuned so that the discrete solution does not "explode" in time. Moreover, since any lattice Boltzmann scheme involves more unknowns q than those of the PDEs to solve N, these schemes feature a certain degree of arbitrariness especially as far as the initialization of the last $q - N$ non-conserved moments $\mathsf{m}_{N+1}(0,\cdot),\dots,\mathsf{m}_q(0,\cdot)$

is concerned, or more generally, for the initialization of the q distribution functions $\mathsf{f}_1(0,\cdot),\dots,\mathsf{f}_q(0,\cdot)$. This topic is treated in Chapter 10. Finally, practical computations must take place on a bounded subset of $\Delta x\,\mathbb{Z}^d$, due to the finite size of the computer memory. This generates a lack of information when trying to implement the stream phase (1.4) at the boundary of the computational domain. Therefore, there is a need to replace this lacking pieces of information with something known, which boils down to enforce essentially numerical boundary conditions. Besides, these numerical boundary conditions on the distribution functions have to be used to enforce the physical boundary conditions pertaining to the continuous system to approximate, which are generally expressed on the N conserved moments. This gap between numerical and physical boundary conditions constitute a stumbling block. A preliminary study on this topic is presented in Chapter 12.

1.7 Our standpoint on lattice Boltzmann methods

In this Chapter 1 and practically in the whole work, we have introduced lattice Boltzmann schemes from an algorithmic standpoint, without trying to derive them [Krüger et al., 2017, Chapter 1 and 3] from the continuous-velocity Boltzmann equation. The reason is that—since the number of discrete velocities is small and kept fixed—the link with the original continuous-velocity Boltzmann equation is very weak. Indeed, lattice Boltzmann methods are not discretizations of the continuous-velocity Boltzmann equation. They can still be recovered as discretizations of a finite-velocities Boltzmann equation, see [Dellar, 2013a]. Still, as for relaxation [Brenier, 1984, Bouchut, 2004] and kinetic [Aregba-Driollet and Natalini, 2000] schemes, based on the so-called "relaxation systems" [Jin and Xin, 1995] (which are indeed discrete-velocities Boltzmann equations if written in the right basis) associated with the system of N conservation law to address, we interpret the extension of the state space more as a way of devising an efficient numerical scheme for the conservation laws than a way of approximating the solution of the "relaxation system". Our approach to the numerical analysis of lattice Boltzmann schemes is uniquely based on the algorithmic description of the methods, without seeing them as peculiar discretizations of an extended relaxation system. This is done—partly—to avoid the difficulties associated with the obligation to link the relaxation time (often denoted ϵ or τ) with the discretization parameter Δt and the relaxation parameters $s_{N+1},\dots,s_q$, and with the fact that lattice Boltzmann scheme need to split the collision and transport phase of the finite-velocities Boltzmann equation. This is a choice specific to our work and secures general results on lattice Boltzmann schemes. Still, we do believe that the point of view featuring finite-velocities Boltzmann equations could be equally valuable, in the future, to elucidate the behavior of lattice Boltzmann methods.

Part I

Lattice Boltzmann schemes on dynamically adapted grids

General introduction

Lattice Boltzmann methods are widespread numerical methods to take a large number of physical phenomena into account, which can feature a broad spectrum of spatial scales and behavior. However, they suffer from the fact that they have originally been designed to work on uniform Cartesian grids, thus lacking flexibility from the geometrical standpoint. Furthermore, they can be particularly demanding in terms of memory occupation, because the number of discrete velocities—and thus of variables to store—can grow steadfastly, especially when two and three-dimensional problems are considered and more complex physics are taken into account. This calls for strategies allowing to reduce the cost of lattice Boltzmann methods, in particular as far as the storage perspective is concerned, while preserving the quality of the numerical solution and computational efficiency.

Aim and structure of Part I

In order to provide an answer to these issues, the aim of Part I is to propose and study—on the one hand—a mesh adaptation strategy, capable of reducing the memory footprint of numerical methods and—on the other hand—adaptive lattice Boltzmann methods to be used on these meshes. The constraints that we impose on our way of proceeding, the mesh adaptation strategy and the way of adapting lattice Boltzmann schemes, as well as several numerical assessments are given in Chapter 2. We conduct a thorough investigation of the method and show that our approach fully complies with the constraints. Then, the numerical strategy we have proposed is investigated with additional detail in Chapter 3, to highlight that it allows to produce adaptive schemes that behave as much as possible as the original lattice Boltzmann scheme. Moreover, our strategy allows us to cure certain issues that typically affect lattice Boltzmann schemes on adaptive grids. Finally, in Chapter 4, we adapt the analysis that has been used to study adaptive lattice Boltzmann scheme to tackle adaptive Finite Volume schemes as well.

Published works

The topics covered in Part I have led to the following publications in peer-reviewed journals.

- [Bellotti et al., 2022d] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022d). Multiresolution-based mesh adaptation and error control for lattice Boltzmann methods with applications to hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 44(4):A2599–A2627.
  This covers the one-dimensional case $d = 1$ for the content of Chapter 2.
- [Bellotti et al., 2022c] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022c). Multidimensional fully adaptive lattice Boltzmann methods with error control based on multiresolution analysis. *Journal of Computational Physics*, 471:111670.
  This covers the two/three-dimensional case $d = 2, 3$ for the content of Chapter 2.
- [Bellotti et al., 2022b] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022b). High accuracy analysis of adaptive multiresolution-based lattice Boltzmann schemes via the equivalent equations. *SMAI Journal of Computational Mathematics*, 8:161–199.
  This features the part of the content of Chapter 3, in particular that of Section 3.1.
- [Bellotti et al., 2022a] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022a). Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps? *Comptes Rendus. Mathématique*, 360:761–769.
  This article includes the material presented in Section 3.2 from Chapter 3.

Chapter 2

Dynamic grid adaptation by multiresolution and adaptive lattice Boltzmann methods

General context and motivation

Steep fronts and shocks are omnipresent in natural phenomena across different disciplines: fluid mechanics, combustion, atmospheric sciences, plasma physics, or biomedical engineering; see [Dumont et al., 2013, Descombes et al., 2014, Duarte et al., 2015, Lecointre, 2022] and references therein. These systems are frequently modeled through very diverse Partial Differential Equations (PDEs), ranging from hyperbolic systems of conservation laws to hyperbolic-parabolic or parabolic systems. The solutions of these PDEs feature small areas where all the variation of the solution is concentrated—such as shocks—and a vast remaining part of the domain where the solution varies smoothly or remains almost constant in large *plateaux*. An effective way of reducing the overall cost of a numerical solver for PDEs consists in devising a strategy to dynamically adapt the spatial discretization to the solution as time advances. This allows to perform fewer operations and limit the memory footprint of the method, while preserving a proper resolution. This comes from the fact that in the areas where the solution varies gently, one can sample the solution coarsely without giving up on the overall quality and accuracy. The discretization of the PDEs can be conducted by relying on several methods. In this thesis, we focus on lattice Boltzmann schemes, a class of widespread numerical methods, first introduced by [McNamara and Zanetti, 1988], see Chapter 1. We aim at defining a numerical strategy which encompasses a very large class of lattice Boltzmann methods.

State of the art

Mesh adaptation

Once a numerical method for the given PDEs is chosen, there exist several strategies for mesh adaptation, both in terms of underlying data structure and refinement strategies. Such strategies can make a crucial difference in terms of time-to-solution and allow scientists to strongly reduce computational cost or reach the solution of large three-dimensional problems on standard machines. In terms of data structure, one can choose either patch-based/block-based [Ray et al., 2007, Narechania et al., 2017] refinement and cell-based/multiresolution [Burstedde et al., 2011, Cohen et al., 2003]. The first kind is generally easier to parallelize, while the second one is more optimal in terms of compression rate. Beyond such choice, adapting the mesh relies on specific refinement/coarsening criteria. There are several possibilities such as feature-based [Guittet et al., 2015], discretization errors [Naddei et al., 2019], as well as *a posteriori* estimates [Wu et al., 1990, Alauzet et al., 2003], Richardson error evaluation [Berger and Oliger, 1984], goal-oriented criteria relying on adjoint evaluation [Narechania et al., 2017], or even optimal sparse sensing [Foti et al., 2020], to cite a few. Since our purpose is to tackle unsteady problems with dynamically evolving meshes and to obtain error control, we focus on the so-called multiresolution approach. With multiresolution analysis—stemming from the pioneer works of [Daubechies, 1988, Mallat, 1989, Cohen et al., 1992]—the discrete solution is decomposed on a local wavelet basis and the corresponding coefficients provide a

precise measure of the local smoothness of the solution, thus supplying essential information on the necessity of refining or coarsening the mesh. The possibility of applying this mechanism to reduce the computational cost of a numerical method has been later studied [Harten, 1994, Harten, 1995, Bihari and Harten, 1997] in the context of Finite Volume methods for conservation laws. The principle is to employ multiresolution to reduce the number of computations during the evaluation of the fluxes at the interfaces of each cell, claiming that they constitute the majority of the computational cost. However, this approach still computes the solution on the full uniform mesh. The possibilities offered by multiresolution have been further exploited by [Cohen et al., 2003], who have developed fully adaptive schemes with solutions updated only on the reduced grid. Thus, multiresolution is not only a way of computing a large number of fluxes more cheaply but also a manner to compute fewer of them. Both these strategies ensure better time-performances than traditional approaches on uniform grids and precise control on the perturbation error, unlike most of the adaptive mesh refinement (AMR) techniques. This strategy has been lately used to tackle various kinds of problems with Finite Volume schemes. We mention parabolic conservation laws by [Roussel et al., 2003], the compressible Navier-Stokes equations in [Bramkamp et al., 2004], the shallow water equations by [Lamby et al., 2005], multicomponent flows in [Coquel et al., 2006], degenerate parabolic equations by [Bürger et al., 2008], and finally the Euler system with a local time-stepping technique again by [Coquel et al., 2010]. Furthermore, this technique has been included in later works to address more complex problems, such as flames [Roussel and Schneider, 2005, Duarte et al., 2013, Descombes et al., 2014, Lecointre, 2022], or by coupling it with other numerical strategies. For this latter use, we mention the works of [Duarte et al., 2012, Dumont et al., 2013, Duarte et al., 2015] and [N'Guessan et al., 2021]. Interestingly, even if comparisons are a difficult task and the results should be interpreted with circumspection, the technique has been compared with the AMR [Deiterding et al., 2016], yielding better compression rates. Though a whole body of literature exists about pointwise multiresolution [Harten, 1993, Chiavassa and Donat, 2001, Forster, 2016, Soni et al., 2017], we decided to focus on volumetric standpoint for multiresolution, because it yields straightforwardly conservative methods.

Adaptive lattice Boltzmann methods

Lattice Boltzmann strategies on adapted grids have been essentially developed on fixed grids which do not evolve in time, in the spirit of [Filippova and Hänel, 1998], see [Lin and Lai, 2000, Kandhai et al., 2000, Yu et al., 2002, Dupuis and Chopard, 2003], where more refined patches are placed according to an a priori knowledge of the flow. Such fixed refinement zones yield difficulties in aeroacoustics resolution, related to the artificial transmission impedance of the refinement interface [Gendre et al., 2017, Feng et al., 2020, Horstmann, 2018], which generate spurious reflected waves. Another strategy is to use an AMR approach with some heuristics to determine the need for refinement in certain areas. In this class, we find the works of [Crouse et al., 2003], using the weighted magnitude of the divergence of the velocity field as regularity indicator; [Eitel-Amor et al., 2013], employing a weighted vorticity and the energy difference with respect to a free flow solution; [Fakhari and Lee, 2014, Fakhari and Lee, 2015], considering the magnitude of the vorticity and its derivative; [Fakhari et al., 2016] using the norm of the gradient of the phase-field. Finally, [Feldhusen et al., 2016] use the magnitude of the gradient for the conserved moments as a refinement criterion. These approaches are highly problem-dependent due to the need to devise criteria adapted to the problem at hand and cannot guarantee control on the perturbation error. Due to the special relation between space and time discretization on the uniform lattice, one should pay special attention to the way of performing the time advancement. On this concern—besides the way of constructing the mesh—we can identify two main trends:

- Methods using local time steps for each level of refinement, thus needing spatial/temporal interpolations and modifications of the collision phase. We can cite those acting on fixed grids [Filippova and Hänel, 1998, Lin and Lai, 2000, Kandhai et al., 2000, Dupuis and Chopard, 2003, Rohde et al., 2006] with nested patches and [Yu et al., 2002] with patches communicating only through edges. [Rohde et al., 2006] have employed the previous approach utilizing volumes to enforce conservation: this is a feature that we retain in our work. The same time stepping approach has been combined with AMR in [Crouse et al., 2003], [Eitel-Amor et al., 2013], and [Feldhusen et al., 2016]. The advantage of this procedure is that it minimizes the number of

time steps but the shortcomings are the need of an adaptation of the collision phase, with possibly singular parameters, and the need of interpolation which calls for massive storage of the solution at the previous time steps.

- Methods using a global time step given by the finest space step and no need to adapt the collision. This is the strategy we shall employ in the work. We can cite [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016]. In these contributions, the authors have utilized a Lax-Wendroff scheme for the stream phase where the adaptation to the local level of refinement is done by modifying the local CFL number. The collision remains untouched and performed locally. This method is simpler to implement, more flexible, and needs less storage than a local time step approach. Still, this approach is used together with heuristic refinement criteria which cannot ensure error control on the numerical solution.

Aims and structure of Chapter 2

The aim of Chapter 2 is to adopt a dynamic mesh adaptation strategy—in particular, multiresolution—and propose a way of reshaping general lattice Boltzmann schemes to be used in this context. The proposed strategy should fulfill the following requirements.

- It must ensure error control with respect to the original lattice Boltzmann scheme, since real applications call for some guarantee on the effect of considering mesh adaptation.
- It must be dynamic in time, because we are interested in problems with shocks, which move as time advances and should therefore be followed by the spatial discretization.
- It has to reduce memory occupation, knowing that lattice Boltzmann methods can be quite expensive from the memory point of view, because of the large number q of discrete velocities.
- It has to be independent of the particular problem and scheme at hand. Indeed, lattice Boltzmann schemes are used in very different contexts. The strategy must be capable of dealing with any multiple-relaxation-times lattice Boltzmann method.

To pursue these ends, Chapter 2 is structured as follows. In Section 2.1, we introduce multiresolution from a theoretical perspective, to understand its basic principles and interests, which go beyond the issues analyzed in this thesis. A preliminary step to employ lattice Boltzmann schemes when adapting meshes with a volumetric point of view is to recast them under a different form, as shown in Section 2.2. On this occasion, we detail the choice of space and time discretizations. Section 2.3 shows how the numerical mesh is coarsened—that is—how cells are eliminated, using multiresolution. This ensures to be able to control the error coming from compressing the mesh by discarding cells. Still, the number of cells in the mesh cannot steadily decrease in time and the possibility of refining has to be taken into account, see Section 2.4. Indeed, this allows to correctly anticipate the singularity formation as well as the propagation of information in the numerical solution. Once the mesh adaptation strategy has been fully detailed, we derive lattice Boltzmann schemes to be used on these moving grids, see Section 2.5. The overall procedure ensures error control—*cf.* Section 2.6—between the solution of the original lattice Boltzmann and its adaptive "*alter ego*". Before testing the proposed strategy, in Section 2.7, we discuss how boundary conditions are taken into account. We finish with a broad number of numerical tests presented in Section 2.8 and we conclude in Section 2.9.

Contents

2.1 Theoretical framework for multiresolution

Let us start by presenting the basic ideas behind multiresolution, which are deeply rooted in the wavelet theory.

2.1.1 A simple example: the Haar wavelet transform

Consider a one dimensional domain $\Omega = [0,1]$ paved with cells

$$C_{\ell,k} = [2^{-\ell}k, 2^{-\ell}(k+1)[,$$

for $\ell \geq \underline{\ell}$ different levels of resolution and an admissible range of indices $k \in [\![0, N_\ell[\![$, where $N_\ell := 2^\ell$ is the number of cells for each level of resolution. Here, $\underline{\ell} \in \mathbb{N}$ is the coarsest level of resolution. For this theoretical presentation, consider a function $f \in L^2([0,1])$, where $(L^2([0,1]), (\cdot,\cdot)_{L^2([0,1])})$ is a Hilbert space if endowed with its standard scalar product. Consider a piecewise constant approximation of the function f over the partition at order ℓ, which reads

$$P_\ell[f](x) := \sum_{k=0}^{N_\ell-1} \underbrace{(f, 2^{\ell/2}\mathbb{1}_{C_{\ell,k}})_{L^2([0,1])}}_{=:a_{\ell,k}[f]} 2^{\ell/2}\mathbb{1}_{C_{\ell,k}}(x).$$

Observe that $P_\ell[f]$ is nothing but the orthogonal projection of f onto the space V_ℓ of square integrable piecewise constant functions over the cells $C_{\ell,k}$. Thanks to the dyadic structure of the cells and their interlocking across levels, one can observe that the spaces are embedded in the sense that

$$V_{\underline{\ell}} \subsetneq V_{\underline{\ell}+1} \subsetneq \cdots \subsetneq V_\ell \subsetneq V_{\ell+1} \subsetneq \ldots, \tag{2.1}$$

with this property being crucial for the following multiresolution analysis. So far, we have not emphasized the fact that we have indeed implicitly selecting a bases for each space V_ℓ, made up of normalized translated box functions $\phi_{\ell,k}$ given by

$$\phi_{\ell,k} = 2^{\ell/2}\mathbb{1}_{C_{\ell,k}} = 2^{\ell/2}\phi(2^\ell \cdot - k),$$

where $\phi = \mathbb{1}_{[0,1[}$ is classically called the "scaling function" or the "box function". These functions make up an orthonormal basis for the space V_ℓ because $(\phi_{\ell,k}, \phi_{\ell,h})_{L^2([0,1])} = \delta_{kh}$. We also introduce the so-called "approximation coefficient" $c_{\ell,k}[f] := (f, \phi_{\ell,k})_{L^2([0,1])} = 2^{-\ell/2} a_{\ell,k}[f]$ at the scale $2^{-\ell}$ for the position $2^{-\ell}k$. It can be proved that

$$\overline{\bigcup_{\ell \geq \underline{\ell}} V_\ell} = L^2([0,1]), \tag{2.2}$$

where the closure is taken with respect to the norm induced by the scalar product of $L^2([0,1])$. Furthermore, this property implies that

$$\lim_{\ell \to +\infty} \|f - P_\ell[f]\|_{L^2([0,1])} = 0, \tag{2.3}$$

using the projection theorem on a closed subset of a Hilbert space together with the density property (2.2). More generally, assuming that $f \in L^p([0,1])$ for $1 \leq p < \infty$ allows to deduce [Duarte, 2011, Chapter 3] that $\lim_{\ell \to +\infty} \|f - P_\ell[f]\|_{L^p([0,1])} = 0$. Also, by virtue of the Heine-Cantor theorem, if $f \in C^0([0,1])$, then $\lim_{\ell \to +\infty} \|f - P_\ell[f]\|_{L^\infty([0,1])} = 0$ as well. These are all nice property of convergence of the projections over spaces of piecewise constant functions, but we still have to introduce the core of multiresolution analysis.

Using the nesting (2.1) and the (2.3), one comes to the telescopic expression

$$f = P_{\underline{\ell}}[f] + \sum_{\ell=\underline{\ell}}^{+\infty} \underbrace{(P_{\ell+1}[f] - P_\ell[f])}_{=:Q_\ell[f]}, \tag{2.4}$$

where the equality holds in the sense of $L^2([0,1])$. This is a multi-scale representation of the function f and involves the functions which are the "details" $Q_\ell[f]$ of f at resolution $2^{-\ell}$. Intuitively, one can decompose f as the sum of its projection over the coarsest level $\underline{\ell}$ (*i.e.* over $V_{\underline{\ell}}$) plus the details for each level $\ell > \underline{\ell}$. It can be easily seen by splitting the associated integral that we have

$$a_{\ell,k}[f] = \frac{1}{2}(a_{\ell+1,2k}[f] + a_{\ell+1,2k+1}[f]).$$

Therefore, in order to reconstruct $P_{\ell+1}[f]$ from the knowledge of $P_\ell[f]$, the function $Q_\ell[f]$ has to oscillate within each cell $C_{\ell,k}$. This suggests to introduce—besides the box function ϕ—an oscillatory profile $\psi = \mathbb{1}_{[0,1/2[} - \mathbb{1}_{[1/2,1[}$, called "Haar wavelet" which shall be used as a basis to represent the details $Q_\ell[f]$. Remark that the following relations hold

$$\psi(x) = \phi(2x) - \phi(2x-1), \qquad \phi(x) = \phi(2x) + \phi(2x-1), \tag{2.5}$$

which also imply, by summation and subtraction

$$\phi(2x) = \frac{1}{2}(\phi(x) + \psi(x)), \qquad \phi(2x-1) = \frac{1}{2}(\phi(x) - \psi(x)).$$

Analogously to the box function ϕ, we introduce the normalized translated Haar wavelet given by $\psi_{\ell,k} := 2^{\ell/2}\psi(2^\ell \cdot - k)$ so that it can be easily seen that

$$Q_\ell[f] = \sum_{k=0}^{N_\ell - 1} \underbrace{(f, \psi_{\ell,k})_{L^2([0,1])}}_{=:d_{\ell,k}[f]} \psi_{\ell,k}, \tag{2.6}$$

where $d_{\ell,k}[f]$ is the wavelet coefficient at the scale $2^{-\ell}$ at position $2^{-\ell}k$. Using the definition of the detail $Q_\ell[f]$, one can easily see that the functions $Q_\ell[f]$ and $P_\ell[f]$ are orthogonal for the $L^2([0,1])$ scalar product: $(Q_\ell[f], P_\ell[f])_{L^2([0,1])} = 0$, which shows that $Q_\ell[f]$ is the orthogonal projection over the space W_ℓ which is the

orthogonal complement of V_ℓ and moreover that the next space $V_{\ell+1}$ is given by

$$V_{\ell+1} = V_\ell \oplus^\perp W_\ell = V_{\ell-1} \oplus^\perp W_{\ell-1} \oplus^\perp W_\ell = \dots \tag{2.7}$$

As a consequence, letting $\ell \to +\infty$ and selecting the minimum level $\underline{\ell} = 0$, one can show that $\{\phi\} \cup \{\psi_{\ell,k}\}_{\ell\geq 0, k\in[\![0,N_\ell[\![}$ is an orthonormal basis of $L^2([0,1])$ called the "Haar system". One can easily prove the two-scale relations

$$\begin{aligned} c_{\ell+1,2k}[f] &= \frac{1}{\sqrt{2}}(c_{\ell,k}[f] + d_{\ell,k}[f]), & c_{\ell+1,2k+1}[f] &= \frac{1}{\sqrt{2}}(c_{\ell,k}[f] - d_{\ell,k}[f]), \\ c_{\ell,k}[f] &= \frac{1}{\sqrt{2}}(c_{\ell+1,2k}[f] + c_{\ell+1,2k+1}[f]), & d_{\ell,k}[f]) &= \frac{1}{\sqrt{2}}(c_{\ell+1,2k}[f] - c_{\ell+1,2k+1}[f]), \end{aligned}$$

meaning that to recover the approximation at a finer level $\ell+1$, one needs both of the decompositions on the basis generated by ϕ and ψ at level ℓ. Quite the opposite, to pass from a finer level $\ell+1$ to a coarser level ℓ, only the decomposition on ϕ is necessary. This gives that

$$P_{\ell+1}[f] = \sum_{k=0}^{N_\ell} c_{\ell,k}[f]\phi_{\ell,k} + \sum_{k=0}^{N_\ell} d_{\ell,k}[f]\psi_{\ell,k}.$$

In fine, given a level $\overline{\ell} \geq \underline{\ell}$, then we have the identity

$$P_{\overline{\ell}}[f] = \underbrace{\sum_{k=0}^{N_{\overline{\ell}}} c_{\overline{\ell},k}[f]\phi_{\overline{\ell},k}}_{\text{on "canonical basis"}} = \underbrace{\sum_{k=0}^{N_{\underline{\ell}}} c_{\overline{\ell},k}[f]\phi_{\underline{\ell},k} + \sum_{\ell=\underline{\ell}}^{\overline{\ell}-1}\sum_{k=0}^{N_\ell-1} d_{\ell,k}[f]\psi_{\ell,k}}_{\text{on wavelet basis}},$$

where we can see the projection at level $\overline{\ell}$ either on the canonical basis or on the wavelet basis. Therefore, there is an isomorphism—called "fast wavelet transform"—between the coefficients $(c_{\overline{\ell},k}[f])_{k\in[\![0,N_{\overline{\ell}}[\![}$ on the canonical basis and the ones $(c_{\underline{\ell},k}[f])_{k\in[\![0,N_{\underline{\ell}}[\![}$ plus $(d_{\ell,k}[f])_{\ell\in[\![\underline{\ell},\overline{\ell}[\![,k\in[\![0,N_\ell[\![}$ on the wavelet basis.

Of course, a very similar decomposition can be proposed for the function f instead that its piecewise constant projection at level $\overline{\ell}$, called $P_{\overline{\ell}}[f]$. This reads, taking $\underline{\ell} = 0$ and the limit $\overline{\ell} \to +\infty$, by the convergence of the orthogonal projection in the $L^2([0,1])$ space

$$f = P_0[f] + \sum_{\ell=0}^{+\infty} Q_\ell[f] = \sum_{\ell=-1}^{+\infty}\sum_{k=0}^{N_\ell-1} d_{\ell,k}[f]\psi_{\ell,k} =: \boldsymbol{d}[f]\cdot\boldsymbol{\psi},$$

merging (2.4) and (2.6) and using $\psi_{-1,k} = \phi_{0,k}$. The wavelet transform is indeed an isometry because of the Parseval's identity

$$\|f\|_{L^2([0,1])} = \|\boldsymbol{d}[f]\|_2,$$

thanks to the fact that $L^2([0,1])$ is a separable Hilbert space—thus isometrically isomorphic to ℓ^2—endowed with an orthonormal basis. From this identity, we see that if we neglect small details, then the norm of the function shall also be modified by a small amount. Moreover, the details decrease in ℓ according to the local regularity of the function f. Assume that $f \in C^1(C_{\ell,k})$, then for every $x \in C_{\ell,k}$, there exist a point $\overline{x} = \overline{x}(x) \in C_{\ell,k}$ such that $f(x) = f'(\overline{x})(x - 2^{-\ell}k)$. We then have, since $C_{\ell+1,2k} \subset C_{\ell,k}$

$$\begin{aligned} |c_{\ell+1,2k}[f]| &= \left|\int_{C_{\ell+1,2k}} 2^{\ell/2} f(x)\mathrm{d}x\right| = \left|\int_{C_{\ell+1,2k}} 2^{\ell/2} f'(\overline{x})(x - 2^{-\ell}k)\mathrm{d}x\right| \\ &\leq 2^{\ell/2}\|f'\|_{L^\infty(C_{\ell+1,2k})}\left|\int_{C_{\ell+1,2k}} (x - 2^{-\ell}k)\mathrm{d}x\right| \lesssim 2^{-3\ell/2}\|f'\|_{L^\infty(C_{\ell+1,2k})}. \end{aligned}$$

The same inequality holds for $C_{\ell+1,2k+1}$. Using (2.5), we gain

$$|d_{\ell,k}[f]| = \left|\frac{1}{\sqrt{2}}(c_{\ell+1,2k}[f] - c_{\ell+1,2k+1}[f])\right| \lesssim 2^{-3\ell/2}\|f'\|_{L^\infty(C_{\ell,k})}.$$

In the manuscript, the symbol $\lesssim$ indicates the inequality $\leq$ up to a multiplicative constant with no dependence

unless specifically stated. This means that for local C^1 regularity, the details decrease with the level ℓ with a specific rate 3/2. Moreover, one can show [Cohen et al., 1992] that if $f \in C^{0,\alpha}(C_{\ell,k})$, then $|d_{\ell,k}[f]| \lesssim 2^{-(\alpha+1/2)\ell}$. Finally notice that the Haar wavelet ψ is such that $(c, \psi_{\ell,k})_{L^2([0,1])} = 0$ for every constant $c \in \mathbb{R}$, which means that is is orthogonal to constant functions.

The limitations of the Haar wavelet are that it is only first-order accurate and it can be sub-optimal in dealing with smooth functions f, due to the fact that the size of the support of ψ is small, indeed contained in $[0,1]$. For these reasons, two solutions can be envisioned. The first one is to use compactly supported orthonormal wavelets of higher order, see [Daubechies, 1988, Daubechies, 1992]. The other approach is the one of the bi-orthogonal wavelets [Cohen et al., 1992, Lemarié-Rieusset, 1996], where the standard L^2 orthogonality of the basis functions is sacrificed in favor of other properties.

2.1.2 Orthonormal wavelets

We present the case on the whole real line $\mathbb{R}$. For a bounded domain Ω, the presentation is a little more involved, see [Cohen et al., 2004]. We first give the definition of multiresolution analysis on $L^2(\mathbb{R})$ as given in [Mallat, 1989, Daubechies, 1992, Kelly et al., 1994]: a sequence $(V_\ell)_{\ell\in\mathbb{N}}$ of closed subspaces of $L^2(\mathbb{R})$ satisfying the nesting

$$V_0 \subset V_1 \subset \cdots \subset V_\ell \subset V_{\ell+1} \subset \ldots,$$

such that

$$\overline{\bigcup_{\ell\in\mathbb{N}} V_\ell} = L^2(\mathbb{R}), \qquad \bigcap_{\ell\in\mathbb{N}} V_\ell = \{0\},$$

is a multiresolution analysis. Moreover, we assume that the following properties hold:

- Every space V_ℓ is a scaling of the central space V_0, that is

$$f \in V_\ell \qquad \Leftrightarrow \qquad f(2^\ell \cdot) \in V_0, \qquad \ell \in \mathbb{N}.$$

- Invariance of the central space V_0 under integer translations, *i.e.*

$$f \in V_0 \qquad \Leftrightarrow \qquad f(\cdot - k) \in V_0, \qquad k \in \mathbb{Z}.$$

- There exists a scaling function $\phi \in V_0$ such that $(\phi_{\ell,k})_{k\in\mathbb{Z}}$ where $\phi_{\ell,k} = 2^{\ell/2}\phi(2^\ell \cdot - k)$ is an orthonormal basis of V_ℓ. This assumption can be weakened by requesting that $(\phi_{\ell,k})_{k\in\mathbb{Z}}$ is just a Riesz basis of V_ℓ, that is that there exist two constants $\underline{C}, \overline{C} > 0$ such that for every $c = (c_k)_{k\in\mathbb{Z}} \in \ell^2(\mathbb{Z})$, then

$$\underline{C}\|c\|_2 \le \left\| \sum_{k\in\mathbb{Z}} c_k \phi(\cdot - k) \right\|_{L^2(\mathbb{R})} \le \overline{C}\|c\|_2, \quad \text{and} \quad \overline{\operatorname{span}(\phi(\cdot - k))_{k\in\mathbb{Z}}} = V_0.$$

  This implied that any function in each V_ℓ has a unique representation on the Riesz basis at hand.

With this, we associate to each space V_ℓ its orthogonal complement W_ℓ in the next space $V_{\ell+1}$, thus analogously to (2.7), such that $V_{\ell+1} = V_\ell \oplus^\perp W_\ell$. The generalization of (2.5) is given by

$$\phi(x) = \sum_{k\in\mathbb{Z}} a_k \phi(2x - k),$$

for a compactly supported $(a_k)_{k\in\mathbb{Z}}$, called "mask". Recall that the Haar wavelet was given by the choice $a_0 = a_1 = 1$. Hence, the basis of each space V_ℓ shall be given by

$$\phi_{\ell,k} = \frac{1}{\sqrt{2}} \sum_{r\in\mathbb{Z}} a_r \phi_{\ell+1,2k+r},$$

and the wavelet ψ shall be constructed by

$$\psi(x) = \sum_{k\in\mathbb{Z}} b_k \phi(2x-k), \qquad \text{with} \qquad b_k = (-1)^k a_{1-k}.$$

The orthonormality comes from the constraints $\sum_{r\in\mathbb{Z}} a_r a_{r+2k} = \delta_{k0}$ for $k \in \mathbb{Z}$. This guarantees that $\psi_{\ell,k} = 2^{\ell/2}\psi(2^\ell \cdot - k)$ are an orthonormal basis of the space W_ℓ, so that, as for the Haar wavelet

$$f = \sum_{k\in\mathbb{Z}} (f,\phi_{0,k})_{L^2(\mathbb{R})}\phi_{0,k} + \sum_{\ell=0}^{+\infty}\sum_{k\in\mathbb{Z}} (f,\psi_{\ell,k})_{L^2(\mathbb{R})}\psi_{\ell,k}.$$

The wavelets are generally built by imposing that a certain number of their moments $M \in \mathbb{N}$ vanish: that is, $(f,\psi_{\ell,k})_{L^2(\mathbb{R})} = 0$ if f is a polynomial of degree at most $M-1$. The constraints to ensure these vanishing moments are

$$\sum_{k\in\mathbb{Z}} a_k = 2, \qquad \sum_{k\in\mathbb{Z}} (-1)^k k^h a_k = 0, \qquad h \in [\![0, M]\!].$$

The decay of the coefficients $(f,\psi_{\ell,k})_{L^2(\mathbb{R})}$ can be obtained with the following procedure. Using [DeVore and Sharpley, 1984, Theorem 3.4], if $f \in W^{M,p}(\mathrm{supp}(\psi_{\ell,k}))$ with $p \in [1,\infty]$, there exists a polynomial π of degree at most $M-1$ such that $\|f-\pi\|_{L^p(\mathrm{supp}(\psi_{\ell,k}))} \lesssim |\mathrm{supp}(\psi_{\ell,k})|^M |f|_{W^{M,p}(\mathrm{supp}(\psi_{\ell,k}))}$. Then, let $1/p + 1/q = 1$

- The wavelet ψ is $L^q(\mathbb{R})$-normalized, then by the Hölder inequality

$$\begin{aligned}|(f,\psi_{\ell,k})_{L^2(\mathbb{R})}| = |(f-\pi,\psi_{\ell,k})_{L^2(\mathbb{R})}| \le \|f-\pi\|_{L^p(\mathbb{R})}\|\psi_{\ell,k}\|_{L^q(\mathbb{R})} &\lesssim |\mathrm{supp}(\psi_{\ell,k})|^M |f|_{W^{M,p}(\mathrm{supp}(\psi_{\ell,k}))} \\ &\lesssim 2^{-\ell M}|f|_{W^{M,p}(\mathrm{supp}(\psi_{\ell,k}))},\end{aligned}$$

  since $|\mathrm{supp}(\psi_{\ell,k})| \lesssim 2^{-\ell}$.

- The wavelet ψ is $L^2(\mathbb{R})$-normalized, then by the Hölder inequality applied to the functions $|\psi_{\ell,k}|^q$ and $\mathbb{1}_{\mathrm{supp}(\psi_{\ell,k})}$ with exponents $2/q$ and its conjugate $2/(2-q)$, we gain

$$\|\psi_{\ell,k}\|^q_{L^q(\mathbb{R})} = \int_{\mathbb{R}} |\psi_{\ell,k}(x)|^q \mathbb{1}_{\mathrm{supp}(\psi_{\ell,k})}(x)\mathrm{d}x \le \|\psi_{\ell,k}\|^2_{L^2(\mathbb{R})}|\mathrm{supp}(\psi_{\ell,k})|^{1-q/2} \lesssim 2^{-\ell(1-q/2)},$$

  hence $\|\psi_{\ell,k}\|_{L^q(\mathbb{R})} \lesssim 2^{-\ell(1/q-1/2)} = 2^{-\ell(1/2-1/p)}$, thus

$$|(f,\psi_{\ell,k})_{L^2(\mathbb{R})}| \le \|f-\pi\|_{L^p(\mathbb{R})}\|\psi_{\ell,k}\|_{L^q(\mathbb{R})} \lesssim 2^{-\ell(M+1/2-1/p)}|f|_{W^{M,p}(\mathrm{supp}(\psi_{\ell,k}))}.$$

To sum up, the previous estimates indicate that the smoothness of the function f directly reverberates on the decay of the coefficients of its wavelet decomposition. Therefore, this representation allows to test the regularity of the encoded data. Similar estimates from and for wavelet coefficients exist for Besov spaces—see [Jaming and Malinnikova, 2016] and references therein—which is unsurprising, for Besov spaces can be characterized using dyadic decompositions of the frequency space in the realm of the Littlewood-Paley theory.

2.1.3 Bi-orthogonal wavelets

We finish this theoretical introduction on bi-orthogonal wavelets, which are constructed over the pair $(\phi,\tilde{\phi})$ of dual scaling functions which satisfy

$$(\phi,\tilde{\phi}(\cdot-k))_{L^2(\mathbb{R})} = \delta_{k0}, \qquad k \in \mathbb{Z},$$

called "bi-orthogonality" which satisfy the generalization of (2.5)

$$\phi(x) = \sum_{k\in\mathbb{Z}} a_k \phi(2x-k), \qquad \tilde{\phi}(x) = \sum_{k\in\mathbb{Z}} \tilde{a}_k \tilde{\phi}(2x-k).$$

The wavelets are analogously defined by

$$\psi(x) = \sum_{k\in\mathbb{Z}} b_k \psi(2x-k), \qquad \tilde{\psi}(x) = \sum_{k\in\mathbb{Z}} \tilde{b}_k \tilde{\psi}(2x-k),$$

where $b_k = (-1)^k a_{1-k}$ and $\tilde{b}_k = (-1)^k \tilde{a}_{1-k}$, with the additional bi-orthogonality constraint

$$(\psi_{\ell,k}, \tilde{\psi}_{r,p})_{L^2(\mathbb{R})} = \delta_{\ell r}\delta_{kp}.$$

This also reads $\sum_{r\in\mathbb{Z}} a_r \tilde{a}_{r+2k} = 2\delta_{k0}$ for $k \in \mathbb{Z}$. In this way

$$\sum_{k\in\mathbb{Z}} (f, \tilde{\phi}_{\ell,k})_{L^2(\mathbb{R})} \phi_{\ell,k}$$

is the non-orthogonal projection of f over a space V_ℓ and

$$\sum_{k\in\mathbb{Z}} (f, \tilde{\psi}_{\ell,k})_{L^2(\mathbb{R})} \psi_{\ell,k}$$

onto the non-orthogonal complement $W_\ell = V_{\ell+1} \cap \tilde{V}_\ell^\perp$. Then it is assumed that $\phi_{0,k} \cup (\psi_{\ell,k})_{\ell\in\mathbb{N}}$ for $k \in \mathbb{Z}$ forms a Riesz basis of $L^2(\mathbb{R})$, hence also $\tilde{\phi}_{0,k} \cup (\tilde{\psi}_{\ell,k})_{\ell\in\mathbb{N}}$ for $k \in \mathbb{Z}$ has the same property. One therefore has

$$f = \sum_{k\in\mathbb{Z}} (f, \phi_{0,k})_{L^2(\mathbb{R})} \tilde{\phi}_{0,k} + \sum_{\ell=0}^{+\infty} \sum_{k\in\mathbb{Z}} (f, \psi_{\ell,k})_{L^2(\mathbb{R})} \tilde{\psi}_{\ell,k},$$
$$f = \sum_{k\in\mathbb{Z}} (f, \tilde{\phi}_{0,k})_{L^2(\mathbb{R})} \phi_{0,k} + \sum_{\ell=0}^{+\infty} \sum_{k\in\mathbb{Z}} (f, \tilde{\psi}_{\ell,k})_{L^2(\mathbb{R})} \psi_{\ell,k}.$$

The advantage of bi-orthogonal wavelets over orthogonal ones is that they guarantee a lot more flexibility—meaning, they allow to span a wider range of possible constructions—still being able to have analogous decay estimates. They also allow to construct symmetric wavelet functions.

Observe that we concentrate on wavelets sampling functions using the box function, yielding a volumetric standpoint. This is the standard approach when employing a wavelet decomposition coupled with Finite Volume methods and we adopt it as well in order to preserve conservation features. However, one has to be aware that the point-wise standpoint also exists [Donoho, 1992, Harten, 1993, Chiavassa and Donat, 2001, Forster, 2016, Soni et al., 2017], where the scaling function is indeed a Dirac mass and thus functions are sampled using point values.

2.2 Lattice Boltzmann schemes on control volumes

We propose a formulation of general Lattice Boltzmann schemes on control volumes, rather than points as normally considered (*cf.* Chapter 1). This choice is done in order to obtain methods preserving the conservation properties of the original method on uniform lattices, even when performing mesh adaptation [Chen, 1998, Rossi et al., 2005, Ubertini and Succi, 2005, Rohde et al., 2006]. Observe that it is most common to interpret the solutions of a lattice Boltzmann scheme in terms of point values (*cf.* Chapter 1) rather than averages on volumes [Dubois and Lallemand, 2008, Caetano et al., 2023].

Consider to work on a bounded domain $\Omega \subset \mathbb{R}^d$ where the dimension of the space is $d = 1, 2, 3$. For the sake of presentation, we consider hyper-cubic domains of the form $\Omega = [0,1]^d$.

2.2.1 Space discretization: nested lattices

The spatial discretization of the domain Ω is done [Harten, 1994, Müller, 2002, Cohen et al., 2003, Hovhannisyan and Müller, 2010, Duarte, 2011] by considering a maximum (respectively, minimum) level of resolution $\overline{\ell} \in \mathbb{N}$ (respectively, $\underline{\ell} \in \mathbb{N}$) with $\overline{\ell} \geq \underline{\ell}$ spanning from the finest to the lower resolution for the considered grid. For a given level $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$, we introduce its distance from the finest level $\overline{\ell}$ indicated by $\Delta\ell := \overline{\ell} - \ell$. Then, one considers

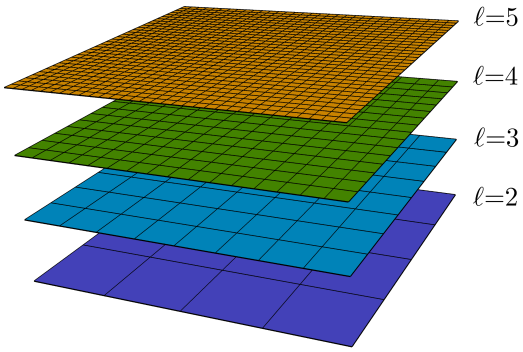


Figure 2.1: Example of nested grids for $d = 2$.

a hierarchy of nested univariate grids made up of cells

$$C_{\ell,\boldsymbol{k}} := \prod_{\alpha=1}^{d} [2^{-\ell} k_\alpha, 2^{-\ell}(k_\alpha + 1)], \tag{2.8}$$

for $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ and admissible range of indices $\boldsymbol{k} \in [\![0, N_\ell[\![^d$, where $N_\ell := 2^\ell$ indicates the number of cells allowed at each level of resolution along each Cartesian direction. The center of a cell $C_{\ell,\boldsymbol{k}}$ is given by $\boldsymbol{x}_{\ell,\boldsymbol{k}} := 2^{-\ell}(\boldsymbol{k} + \mathbf{1/2})$. These cells are nested and the union of the so-called "children" cells makes up the "parent" cell, *i.e.*

$$\bigcup_{\boldsymbol{\delta} \in \Sigma} C_{\ell+1, 2\boldsymbol{k}+\boldsymbol{\delta}} = C_{\ell,\boldsymbol{k}}, \qquad \text{where} \qquad \Sigma := [\![0,1]\!]^d, \tag{2.9}$$

for $\boldsymbol{k} \in [\![0, N_\ell[\![^d$ and $\ell \in [\![\underline{\ell}, \overline{\ell}[\![$, see also Figure 2.1. Although this nested structure can be seen in terms of trees (binary trees for $d = 1$, quadtrees for $d = 2$ and eventually octrees for $d = 3$) and this point of view is widespread in the literature [Burstedde et al., 2011], we shall not explicitly construct it in what follows to avoid their intrinsically recursive nature, *cf.* Part II. The edge length of each cell at level $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ is indicated by $\Delta x_\ell := 2^{-\ell}$—so that $|C_{\ell,\boldsymbol{k}}|_d = 2^{-d\ell}$. Moreover, due to its particular role (*cf.* Section 2.2.2), we indicate $\Delta x = \Delta x_{\overline{\ell}}$ the length of the edge at the finest level $\overline{\ell}$.

2.2.2 Time discretization: global time-step

Having introduced a discretization of the space domain Ω, we can now turn to the discretization of the time domain. In any lattice Boltzmann scheme on a uniform lattice, the space-step Δx and time-step Δt are linked through a specific scaling $\Delta t = \Lambda(\Delta x)$ where $\lim_{\xi \to 0} \Lambda(\xi) = 0$ and $\Lambda > 0$, see Section 1.2. Now that several spatial scales, spanned by $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$, have been introduced, one can choose to utilize a global time-step across all grid levels ℓ [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016], or to employ local time-steps for each level of resolution [Filippova and Hänel, 1998, Rohde et al., 2006].

The latter strategy aims at keeping the speed of propagation of information (*i.e.*, the ratio between local time-step and space-step) constant throughout $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$. Drawbacks are that this approach compels to modify the collision phase of the method—introduced in Section 1.4.1—to recover the same numerical diffusion. Controlling this numerical dissipation is crucial to model the fluid viscosity when using a D_2Q_9 scheme with three conserved moments [Lallemand and Luo, 2000] under acoustic scaling. Moreover, this approach needs to synchronize the different levels with time interpolations and rescalings of the distribution functions.

To avoid these complications and to allow us to employ our approach regardless of the structure of the lattice Boltzmann at hand (choice of moments, collision phase, *etc.*) and more importantly to secure error control, we consider only one $\Delta t = \Lambda(\Delta x)$ dictated by the finest spatial scale Δx corresponding to $\overline{\ell}$, following [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016]. The notable advantages of this approach are that relaxation parameters do not need to be scaled and thus one does not have to deal with singular values (unlike [Filippova and

Hänel, 1998]) and multiple-relaxation-times are naturally handled. Furthermore, temporal interpolations across grids are avoided and the solution at sub time-steps do not need to be stored.

2.2.3 Time integration

Since the numerical spatial grid is made up of volumes, we have to propose lattice Boltzmann schemes to work on this kind of discretization. With space and time discretizations at hand, we can perform the first step to obtain the reference (meaning at the finest level of resolution) lattice Boltzmann method on volumes.

Volumetric lattice Boltzmann schemes could be obtained [Rohde et al., 2006] by simply taking the point-wise lattice Boltzmann scheme and apply it as if the point-wise distribution functions were averages on the corresponding cell. Still, even if we do not rely much on the kinetic interpretation of lattice Boltzmann schemes—as we will emphasize in Part III—this can be the starting point of the derivation. Consider the discrete-velocity Boltzmann equation [Broadwell, 1964, Platkowski and Illner, 1988, He et al., 1998] with multiple-relaxation-times

$$\partial_t f^j(t,\boldsymbol{x}) + \boldsymbol{\xi}_j \cdot \nabla_{\boldsymbol{x}} f^j(t,\boldsymbol{x}) = -\sum_{i=1}^{q} \omega_{ji}(f^i - f^{\mathrm{eq},i})(t,\boldsymbol{x}), \qquad j \in [\![1,q]\!], \tag{2.10}$$

where the matrix $\boldsymbol{\omega} \in \mathcal{M}_q(\mathbb{R})$ contains the reciprocal of the relaxation times. The discrete velocities $(\boldsymbol{\xi}_j)_{j\in[\![1,q]\!]} \subset \mathbb{R}^d$ introduced in Section 1.3 can be chosen following the guidelines of [Krüger et al., 2017, Section 3.4] Here, for the sake of deriving a scheme, we forget about the fact that Ω is bounded, hence $\boldsymbol{x} \in \mathbb{R}^d$ and $\boldsymbol{k} \in \mathbb{Z}^d$. Following [He et al., 1998, Dellar, 2003], we evaluate (2.10) on the characteristics of each discrete velocity and integrate in time

$$\begin{aligned}\int_t^{t+\Delta t} (\partial_t f^j + \boldsymbol{\xi}_j \cdot \nabla_{\boldsymbol{x}} f^j)(s,\boldsymbol{x}+s\boldsymbol{\xi}_j)\mathrm{d}s &= -\sum_{i=1}^{q} \omega_{ji} \int_t^{t+\Delta t} (f^i - f^{\mathrm{eq},i})(s,\boldsymbol{x}+s\boldsymbol{\xi}_j)\mathrm{d}s \\ &= f^j(t+\Delta t,\boldsymbol{x}+\Delta t\boldsymbol{\xi}_j) - f^j(t,\boldsymbol{x}) \\ &= -\frac{\Delta t}{2}\sum_{i=1}^{q} \omega_{ji}\big((f^i - f^{\mathrm{eq},i})(t+\Delta t,\boldsymbol{x}+\Delta t\boldsymbol{\xi}_j) + (f^i - f^{\mathrm{eq},i})(t,\boldsymbol{x})\big),\end{aligned}$$

for $j \in [\![1,q]\!]$, where the integral of the collision term is approximated using a trapezoidal rule. The method is kept explicit by setting $\tilde{f}^j = f^j + \Delta t/2 \sum_{i=1}^{q} \omega_{ji}(f^i - f^{\mathrm{eq},i})$, yielding

$$\tilde{f}^j(t+\Delta t,\boldsymbol{x}+\Delta t\boldsymbol{\xi}_j) - \tilde{f}^j(t,\boldsymbol{x}) = -\Delta t \sum_{i=1}^{q} \omega_{ji}(f^i - f^{\mathrm{eq},i})(t,\boldsymbol{x}), \qquad j \in [\![1,q]\!].$$

We are left to rewrite the right-hand side otherwise. By construction $\tilde{\boldsymbol{f}} - \boldsymbol{f}^{\mathrm{eq}} = (\boldsymbol{I} + \Delta t/2\boldsymbol{\omega})(\boldsymbol{f} - \boldsymbol{f}^{\mathrm{eq}})$, hence $(\boldsymbol{f} - \boldsymbol{f}^{\mathrm{eq}}) = (\boldsymbol{I} + \Delta t/2\boldsymbol{\omega})^{-1}(\tilde{\boldsymbol{f}} - \boldsymbol{f}^{\mathrm{eq}})$, providing

$$\tilde{f}^j(t+\Delta t,\boldsymbol{x}+\Delta t\boldsymbol{\xi}_j) - \tilde{f}^j(t,\boldsymbol{x}) = -\Delta t\Big(\boldsymbol{\omega}\Big(\boldsymbol{I} + \frac{\Delta t}{2}\boldsymbol{\omega}\Big)^{-1}(\tilde{\boldsymbol{f}} - \boldsymbol{f}^{\mathrm{eq}})\Big)_j(t,\boldsymbol{x}), \qquad j \in [\![1,q]\!]. \tag{2.11}$$

This is the semi-discretized numerical scheme. We now drop the tilde for the sake of notation.

2.2.4 Spatial averages

We take the average of (2.11) over any $C_{\overline{\ell},\overline{\boldsymbol{k}}}$ at the finest level $\overline{\ell}$ with $\overline{\boldsymbol{k}} \in \mathbb{Z}^d$, knowing that at the very end, we look for a scheme evolving cell averages.

$$\frac{1}{\Delta x^d}\int_{C_{\overline{\ell},\overline{\boldsymbol{k}}}} f^j(t+\Delta t,\boldsymbol{x}+\Delta t\boldsymbol{\xi}_j)\mathrm{d}\boldsymbol{x} - \frac{1}{\Delta x^d}\int_{C_{\overline{\ell},\overline{\boldsymbol{k}}}} f^j(t,\boldsymbol{x})\mathrm{d}\boldsymbol{x} = -\frac{\Delta t}{\Delta x^d}\Big(\boldsymbol{\omega}\Big(\boldsymbol{I} + \frac{\Delta t}{2}\boldsymbol{\omega}\Big)^{-1}\int_{C_{\overline{\ell},\overline{\boldsymbol{k}}}} (\boldsymbol{f} - \boldsymbol{f}^{\mathrm{eq}})(t,\boldsymbol{x})\mathrm{d}\boldsymbol{x}\Big)_j,$$

with $j \in [\![1,q]\!]$. Performing a change of variable in the first integral, indicating averages with a bar and approximating the average of equilibria with the equilibria evaluated on the averages yields

$$\overline{\mathrm{f}}^j_{\overline{\ell},\overline{\boldsymbol{k}}+\boldsymbol{c}_j}(t+\Delta t) = \overline{\mathrm{f}}^j_{\overline{\ell},\overline{\boldsymbol{k}}}(t) - \Delta t\Big(\boldsymbol{\omega}\Big(\boldsymbol{I} + \frac{\Delta t}{2}\boldsymbol{\omega}\Big)^{-1}\Big(\overline{\mathbf{f}}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) - \boldsymbol{f}^{\mathrm{eq}}\big(\overline{\mathrm{f}}^1_{\overline{\ell},\overline{\boldsymbol{k}}}(t),\dots,\overline{\mathrm{f}}^q_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\big)\Big)\Big)_j, \qquad j \in [\![1,q]\!], \tag{2.12}$$

which is fully discrete in time and space, as the point-wise version of Chapter 1 was. Observe that when the equilibria are non-linear functions of their arguments, the average of the equilibria does not equal the equilibria computed on the averages.

2.2.5 Numerical algorithm: collide and stream

To practically implement the scheme given by (2.12) to evolve the solution from time $t \in \Delta t\mathbb{N}$ to $t+\Delta t \in \Delta t\mathbb{N}$, one observes that we can separate a local collision phase and a non-local stream phase.

2.2.5.1 Collision

The collision phase reads

$$\bar{\mathbf{f}}^{\star}_{\ell,\overline{\boldsymbol{k}}}(t) = \Big(\boldsymbol{I} - \Delta t\boldsymbol{\omega}\Big(\boldsymbol{I}+\frac{\Delta t}{2}\boldsymbol{\omega}\Big)^{-1}\Big)\bar{\mathbf{f}}_{\ell,\overline{\boldsymbol{k}}}(t) + \Delta t\boldsymbol{\omega}\Big(\boldsymbol{I}+\frac{\Delta t}{2}\boldsymbol{\omega}\Big)^{-1}\boldsymbol{f}^{\mathrm{eq}}\big(\bar{\mathsf{f}}^{1}_{\ell,\overline{\boldsymbol{k}}}(t),\dots,\bar{\mathsf{f}}^{q}_{\ell,\overline{\boldsymbol{k}}}(t)\big).$$

The change of basis through the moment matrix $\boldsymbol{M} \in \mathrm{GL}_q(\mathbb{R})$ is introduced to diagonalize $\Delta t\boldsymbol{\omega}(\boldsymbol{I}+\Delta t/2\boldsymbol{\omega})^{-1}$, yielding

$$\boldsymbol{M}^{-1}\Big(\Delta t\boldsymbol{\omega}\Big(\boldsymbol{I}+\frac{\Delta t}{2}\boldsymbol{\omega}\Big)^{-1}\Big)\boldsymbol{M} = \boldsymbol{S} = \mathrm{diag}(s_1,\dots,s_N,s_{N+1},\dots,s_q).$$

Moreover, one takes distributions at equilibrium that are functions only of the first N moments, being conserved, where the moments are given by $\overline{\mathbf{m}} = \boldsymbol{M}\bar{\mathbf{f}}$ and $\boldsymbol{m}^{\mathrm{eq}} = \boldsymbol{M}\boldsymbol{f}^{\mathrm{eq}}$. This yields

$$\overline{\mathbf{m}}^{\star}_{\ell,\overline{\boldsymbol{k}}}(t) = (\boldsymbol{I}-\boldsymbol{S})\overline{\mathbf{m}}_{\ell,\overline{\boldsymbol{k}}}(t) + \boldsymbol{S}\boldsymbol{m}^{\mathrm{eq}}\big(\overline{\mathsf{m}}^{1}_{\ell,\overline{\boldsymbol{k}}}(t),\dots,\overline{\mathsf{m}}^{N}_{\ell,\overline{\boldsymbol{k}}}(t)\big), \tag{2.13}$$

which is (1.1) applied to cell averages instead of point values.

2.2.5.2 Stream

This is followed by the stream phase which reads

$$\bar{\mathsf{f}}^{j}_{\ell,\overline{\boldsymbol{k}}+\boldsymbol{c}_j}(t+\Delta t) = \bar{\mathsf{f}}^{j,\star}_{\ell,\overline{\boldsymbol{k}}}(t), \qquad j \in [\![1,q]\!],$$

which again reads as (1.4) applied to cell averages instead of point values upon performing a change of indices:

$$\bar{\mathsf{f}}^{j}_{\ell,\overline{\boldsymbol{k}}}(t+\Delta t) = \bar{\mathsf{f}}^{j,\star}_{\ell,\overline{\boldsymbol{k}}-\boldsymbol{c}_j}(t), \qquad j \in [\![1,q]\!]. \tag{2.14}$$

This gives the lattice Boltzmann that we employ as "reference scheme" [Hovhannisyan and Müller, 2010] on the reference uniform finest grid, being the starting point to devise an adaptive scheme on moving grids. In what follows, the operator associated with this scheme is denoted $\mathbf{E}$, so that for any $t \in \Delta t\mathbb{N}$

$$(\bar{\mathsf{f}}^{j}_{\ell,\overline{\boldsymbol{k}}}(t+\Delta t))_{\substack{\overline{\boldsymbol{k}}\in[\![0,N_\ell[\![^d \\ j\in[\![1,q]\!]}} = \mathbf{E}\,(\bar{\mathsf{f}}^{j}_{\ell,\overline{\boldsymbol{k}}}(t))_{\substack{\overline{\boldsymbol{k}}\in[\![0,N_\ell[\![^d \\ j\in[\![1,q]\!]}}.$$

Observe that [Dubois and Lallemand, 2008] have proposed an interpretation of the point-wise lattice Boltzmann scheme (1.1)/(1.4) in Chapter 1—and in particular its conserved moments—from the point of view of Finite Volume schemes. This is achieved by constructing *ad hoc* control volumes which generally do not coincide with those—namely (2.8)—for (2.13)/(2.14)

2.3 Static mesh adaptation using multiresolution

In Section 2.2, we have presented the volumetric lattice Boltzmann methods on a uniform grid and the framework of nested dyadic grids, without saying how to locally adapt the latter. We here start to answer this question, adapting grids using multiresolution applied to the numerical solution at each time step. This provides a measure of

the local smoothness of the numerical solution, allowing to eliminate cells with error control. Since this procedure is still static with respect to the time step $t \in \Delta t\mathbb{N}$, we omit the time for the sake of clarity. Moreover, as long as the index of the distribution $j \in [\![1, q]\!]$ does not matter, we do not list it.

2.3.1 Projection and prediction operators

We define the projection and prediction operator, allowing to propagate information between different consecutive levels of grid. These operators are respectively the equivalent of the restriction and prolongation operators in the context of multigrid methods.

2.3.1.1 Projection operator

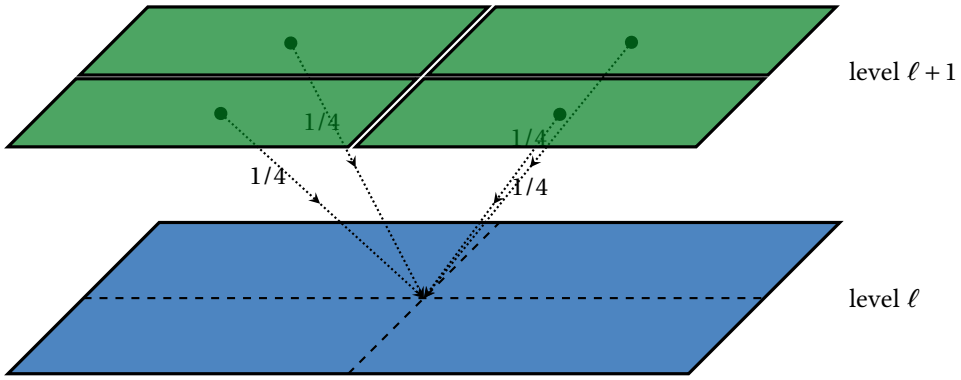


Figure 2.2: Illustration of the action of the projection operator in the context of $d = 2$. The cell average on the cell at level ℓ is reconstructed by taking the average of the values on its four children at level $\ell + 1$.

We start by the projection operator, which takes information at a certain level of resolution $\ell+1$ and transforms it into information on a coarser level ℓ as illustrated in Figure 2.2.

Definition 2.3.1: Projection operator

Let $\ell \in [\![\underline{\ell}, \overline{\ell}[\![$. The projection operator $\boldsymbol{P}_\triangledown : \mathbb{R}^{2d} \to \mathbb{R}$ taking data at level $\ell + 1$, yielding data on level ℓ is defined by

$$\bar{\mathsf{f}}_{\ell,\boldsymbol{k}} = \boldsymbol{P}_\triangledown\big((\bar{\mathsf{f}}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}})_{\boldsymbol{\delta}\in\Sigma}\big) = \frac{1}{2^d}\sum_{\boldsymbol{\delta}\in\Sigma} \bar{\mathsf{f}}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}},$$

for $\boldsymbol{k} \in [\![0, N_\ell[\![^d$.

Given a cell, the projection operator is fully local and is the unique operator passing information from level $\ell+1$ to ℓ exactly conserving the average [Cohen et al., 2003, Duarte, 2011]. This holds because it takes the average of the values defined on children cells which are nested inside their parent, *cf.* (2.9).

2.3.1.2 Prediction operator

The prediction operator acts in the opposite direction, thus transforming information known on a coarse level ℓ to one on a finer level $\ell + 1$. This is the crucial ingredient of multiresolution and it is not uniquely defined, because there are infinite ways of reconstructing the lacking pieces of information "destroyed" by the projection operator. In order to ensure the feasibility of the whole procedure and enforce conservation, which one expects from a lattice Boltzmann method, some constraints [Cohen et al., 2003] have to be imposed.

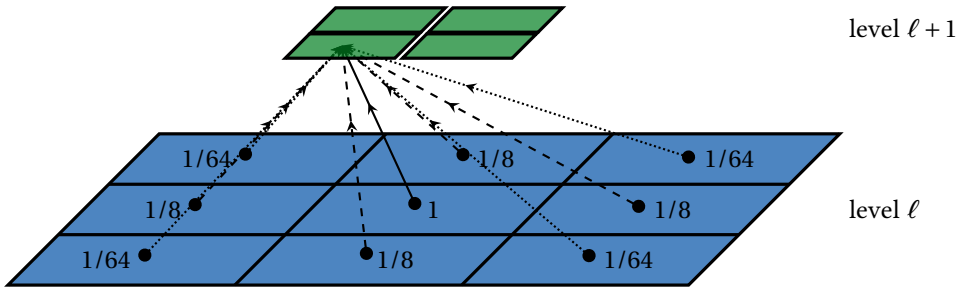


Figure 2.3: Illustration of the action of the prediction operator in the context of $d = 2$ for $\gamma = 1$. We indicate the absolute value of the weight for each cell in the prediction stencil. The parent cell (weight 1) is connected through a continuous arrow, whereas neighbors along the axis (weight $\pm 1/8$) with a dashed arrow and along the diagonals (weight $\pm 1/64$) with a dotted arrow.

Table 2.1: Coefficients used in the prediction operators obtained by polynomial centered reconstruction of the datum. Taken from [Harten, 1994, (1.7a)], [Müller, 2002, Table 2.1 and 2.2] [Duarte, 2011, Table 7.1].

γ	ψ_1	ψ_2	ψ_3	ψ_4
0	-	-	-	-
1	$-1/8$	-	-	-
2	$-11/64$	$3/128$	-	-
3	$-201/1024$	$11/256$	$-5/1024$	-
4	$-3461/16384$	$949/16384$	$-185/16384$	$35/32768$

Definition 2.3.2: Prediction operator

Let $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$. The prediction operator $\boldsymbol{P}_\triangle : \mathbb{R}^{1+w} \to \mathbb{R}^{2d}$ taking data at level ℓ, yielding guessed data on level $\ell+1$ (denoted by hats) is defined by

$$(\hat{\bar{\mathsf{f}}}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}})_{\boldsymbol{\delta}\in\Sigma} = \boldsymbol{P}_\triangle\big((\bar{\mathsf{f}}_{\ell,\boldsymbol{\sigma}})_{\boldsymbol{\sigma}\in N_{\ell,\boldsymbol{k}}}\big),$$

with $\boldsymbol{k} \in [\![0, N_\ell]\!]^d$, fulfilling the following requirements.

- The operator is local, namely the predicted value on any cell $C_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}$ for $\boldsymbol{\delta} \in \Sigma$ depends on the values on $1+w$ cells at level ℓ in the neighborhood of the parent cell $C_{\ell,\boldsymbol{k}}$.
- The operator is consistent with the projection operator from Definition 2.3.1, namely

$$(\boldsymbol{P}_\triangledown \circ \boldsymbol{P}_\triangle)(\bar{\mathsf{f}}_{\ell,\boldsymbol{k}}, \underbrace{\cdots,\cdots}_{w \text{ terms}}) = \bar{\mathsf{f}}_{\ell,\boldsymbol{k}}. \tag{2.15}$$

The second point in Definition 2.3.2 guarantees that the up and down operations between levels conserve the averages. For this reason, the parent cell necessarily belongs to the prediction stencil of its children (this is the 1 in $1+w$). Also observe that Definition 2.3.2 does not compel to consider linear operators, even if this is the choice in many preceding works [Harten, 1994, Harten, 1995, Müller, 2002, Cohen et al., 2003, Hovhannisyan and Müller, 2010, Duarte, 2011, N'Guessan et al., 2021], to cite a few, and in this thesis. This choice has several advantages, among which we mention the possibility of implementing the prediction operator efficiently, see Section 2.5.4.1 and [Cohen et al., 2003, Section 3.5]. Moreover, this choice shall also yield very important numerical properties which will be thoroughly assessed in Chapter 3.

To provide these linear prediction operators, let $\gamma \in \mathbb{N}$ be the number of neighbors in each Cartesian direction to be used by the prediction operator. We consider the prediction operators generated by polynomial centered reconstructions ($d = 1$) and their generalizations ($d > 1$) by tensor product [Bihari and Harten, 1997]. This is [Duarte, 2011, Section 7.1.2]

- For $d = 1$, we utilize

$$\hat{\bar{f}}_{\ell+1,2k+\delta} = \bar{f}_{\ell,k} + (-1)^{\delta} Q_1^{\gamma}(k;\bar{\mathbf{f}}_{\ell}), \quad \text{where} \quad Q_1^{\gamma}(k;\bar{\mathbf{f}}_{\ell}) := \sum_{\sigma=1}^{\gamma} \psi_{\sigma}(\bar{f}_{\ell,k+\sigma} - \bar{f}_{\ell,k-\sigma}), \tag{2.16}$$

  for $\delta \in \Sigma = [\![0,1]\!]$, where the weights $(\psi_\sigma)_{\sigma\in[\![1,\gamma]\!]}$ are given on Table 2.1.

- For $d = 2$, the idea is the same as for $d = 1$ on the x and y axes plus a tensor product along the diagonals. This gives

$$\begin{aligned}\hat{\bar{f}}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}} &= \bar{f}_{\ell,\boldsymbol{k}} + (-1)^{\delta_1} Q_1^{\gamma}(k_1;\bar{\mathbf{f}}_{\ell,(\cdot,k_2)}) + (-1)^{\delta_2} Q_1^{\gamma}(k_2;\bar{\mathbf{f}}_{\ell,(k_1,\cdot)}) - (-1)^{\delta_1+\delta_2} Q_2^{\gamma}(\boldsymbol{k};\bar{\mathbf{f}}_{\ell}), \quad \text{where} \\ Q_2^{\gamma}(\boldsymbol{k};\bar{\mathbf{f}}_{\ell}) &:= \sum_{\sigma_1=1}^{\gamma}\sum_{\sigma_2=1}^{\gamma} \psi_{\sigma_1}\psi_{\sigma_2}(\bar{f}_{\ell,(k_1+\sigma_1,k_2+\sigma_2)} - \bar{f}_{\ell,(k_1-\sigma_1,k_2+\sigma_2)} - \bar{f}_{\ell,(k_1+\sigma_1,k_2-\sigma_2)} + \bar{f}_{\ell,(k_1-\sigma_1,k_2-\sigma_2)}),\end{aligned} \tag{2.17}$$

  for $\boldsymbol{\delta} \in \Sigma = [\![0,1]\!]^2$. The way (2.17) acts is illustrated in Figure 2.3 for the important case $\gamma = 1$.

- For $d = 3$, the idea is the same as for $d = 2$ plus a tensor product along the three-dimensional diagonals.

$$\begin{aligned}\hat{\bar{f}}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}} &= \bar{f}_{\ell,\boldsymbol{k}} + (-1)^{\delta_1} Q_1^{\gamma}(k_1;\bar{\mathbf{f}}_{\ell,(\cdot,k_2,k_3)}) + (-1)^{\delta_2} Q_1^{\gamma}(k_2;\bar{\mathbf{f}}_{\ell,(k_1,\cdot,k_3)}) + (-1)^{\delta_3} Q_1^{\gamma}(k_3;\bar{\mathbf{f}}_{\ell,(k_1,k_2,\cdot)}) \\ &- (-1)^{\delta_1+\delta_2} Q_2^{\gamma}((k_1,k_2);\bar{\mathbf{f}}_{\ell,(\cdot,\cdot,k_3)}) - (-1)^{\delta_1+\delta_3} Q_2^{\gamma}((k_1,k_3);\bar{\mathbf{f}}_{\ell,(\cdot,k_2,\cdot)}) - (-1)^{\delta_2+\delta_3} Q_2^{\gamma}((k_2,k_3);\bar{\mathbf{f}}_{\ell,(k_1,\cdot,\cdot)}) \\ &+ (-1)^{\delta_1+\delta_2+\delta_3} Q_3^{\gamma}(\boldsymbol{k},\bar{\mathbf{f}}_{\ell}),\end{aligned} \tag{2.18}$$

  where

$$\begin{aligned}Q_3^{\gamma}(\boldsymbol{k},\bar{\mathbf{f}}_{\ell}) &:= \sum_{\sigma_1=1}^{\gamma}\sum_{\sigma_2=1}^{\gamma}\sum_{\sigma_3=1}^{\gamma} \psi_{\sigma_1}\psi_{\sigma_2}\psi_{\sigma_3} \\ &\times (\bar{f}_{\ell,(k_1+\sigma_1,k_2+\sigma_2,k_3+\sigma_3)} - \bar{f}_{\ell,(k_1-\sigma_1,k_2+\sigma_2,k_3+\sigma_3)} - \bar{f}_{\ell,(k_1+\sigma_1,k_2-\sigma_2,k_3+\sigma_3)} - \bar{f}_{\ell,(k_1+\sigma_1,k_2+\sigma_2,k_3-\sigma_3)} \\ &+ \bar{f}_{\ell,(k_1-\sigma_1,k_2-\sigma_2,k_3+\sigma_3)} + \bar{f}_{\ell,(k_1-\sigma_1,k_2+\sigma_2,k_3-\sigma_3)} + \bar{f}_{\ell,(k_1+\sigma_1,k_2-\sigma_2,k_3-\sigma_3)} - \bar{f}_{\ell,(k_1-\sigma_1,k_2-\sigma_2,k_3-\sigma_3)}),\end{aligned}$$

  for $\boldsymbol{\delta} \in \Sigma = [\![0,1]\!]^3$.

These operators satisfy the requirements highlighted in Definition 2.3.2, namely the locality and the consistency with the projection operator.

Remark 2.3.1 (Maximum principle). *When $\gamma > 0$, these operators are not convex combinations of the data, so we cannot expect any maximum principle on the predicted values to be preserved. This issue has been addressed in [Pan et al., 2018], targeting specific applications in the context of Finite Volume methods.*

We present the way these prediction operators are retrieved, because these procedures will be useful in Section 2.5.3 and Chapter 3. As previously pointed out, the derivation starts from $d = 1$. Let $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ and $k \in \mathbb{Z}$, since we do not care about boundaries. We consider a local reconstruction polynomial of degree 2γ centered around $C_{\ell,k}$

$$\pi_{\ell,k}(x) = \sum_{m=0}^{2\gamma} \pi_{\ell,k}^m \left(\frac{x - x_{\ell,k}}{\Delta x_\ell}\right)^m. \tag{2.19}$$

The coefficients $(\pi_{\ell,k}^m)_{m\in[\![0,2\gamma]\!]} \subset \mathbb{R}$ are found by enforcing that the averages of the local reconstruction polynomial on the $1 + 2\gamma$ cells surrounding $C_{\ell,k}$ are exactly recovered ([Hovhannisyan and Müller, 2010, Equation (4.1)] and [Duarte, 2011, Equation (3.83)], analogously consider [Müller, 2002, Equation (2.35)]). This provides the following constraints

$$\frac{1}{\Delta x_\ell}\int_{C_{\ell,k+\delta}} \pi_{\ell,k}(x)\mathrm{d}x = \bar{f}_{\ell,k+\delta}, \quad \text{for} \quad \delta \in [\![-\gamma,\gamma]\!]$$

and yields a system with matrix $\boldsymbol{R} \in \mathrm{GL}_{2\gamma+1}(\mathbb{R})$ reading as

$$\boldsymbol{R}(\pi_{\ell,k}^m)_{m\in[\![0,2\gamma]\!]} = (\bar{f}_{\ell,k+\delta})_{\delta\in[\![-\gamma,\gamma]\!]}. \tag{2.20}$$

When the coefficients $(\pi^m_{\ell,k})_{m\in[\![0,2\gamma]\!]}$ depending on $(\bar{f}_{\ell,k+\delta})_{\delta\in[\![-\gamma,\gamma]\!]}$ are found, the local reconstruction polynomial is averaged on $C_{\ell+1,2k+\delta}$ with $\delta\in\Sigma$ to obtain the predicted value:

$$\hat{\bar{f}}_{\ell+1,2k+\delta}=\frac{1}{\Delta x_{\ell+1}}\int_{C_{\ell+1,2k+\delta}}\pi_{\ell,k}(x)\mathrm{d}x.$$

Example 2.3.1 ($\gamma=1$). *To provide an easy example, consider $\gamma=1$. One obtains*

$$\boldsymbol{R}=\begin{bmatrix}1&-1&13/12\\1&0&1/12\\1&1&13/12\end{bmatrix},\quad \textit{whence}\quad \boldsymbol{R}^{-1}=\begin{bmatrix}-1/24&13/12&-1/24\\-1/2&0&1/2\\1/2&-1&1/2\end{bmatrix}.$$

This provides the local reconstruction polynomial given by

$$\begin{aligned}\pi_{\ell,k}(x)=\Big(-\frac{1}{24}\bar{f}_{\ell,k-1}+\frac{13}{12}\bar{f}_{\ell,k}-\frac{1}{24}\bar{f}_{\ell,k+1}\Big)+\Big(-\frac{1}{2}\bar{f}_{\ell,k-1}+\frac{1}{2}\bar{f}_{\ell,k+1}\Big)\Big(\frac{x-x_{\ell,k}}{\Delta x_\ell}\Big)\\+\Big(\frac{1}{2}\bar{f}_{\ell,k-1}-\bar{f}_{\ell,k}+\frac{1}{2}\bar{f}_{\ell,k+1}\Big)\Big(\frac{x-x_{\ell,k}}{\Delta x_\ell}\Big)^2.\end{aligned}\tag{2.21}$$

Taking its averages on the children of $C_{\ell,k}$ yields

$$\hat{\bar{f}}_{\ell+1,2k+\delta}=\bar{f}_{\ell,k}-\frac{(-1)^\delta}{8}(\bar{f}_{\ell,k+1}-\bar{f}_{\ell,k-1}).\tag{2.22}$$

Thanks to the way of constructing the prediction operator, we have an accuracy result on polynomial functions. Its proof directly comes from the previous development.

Proposition 2.3.1: Prediction accuracy

The prediction operator $\boldsymbol{P}_\triangle$ defined by (2.16) is accurate of order $2\gamma+1$, that is, it is exact when $\bar{f}$ stems from the averages of polynomials up to degree 2γ. This reads

$$\frac{1}{\Delta x_{\ell+1}}\int_{C_{\ell+1,2k+\delta}}x^h\mathrm{d}x=\frac{1}{\Delta x_\ell}\int_{C_{\ell,k}}x^h\mathrm{d}x+(-1)^\delta\sum_{\sigma=1}^{\gamma}\psi_\sigma\Big(\frac{1}{\Delta x_\ell}\int_{C_{\ell,k+\sigma}}x^h\mathrm{d}x-\frac{1}{\Delta x_\ell}\int_{C_{\ell,k-\sigma}}x^h\mathrm{d}x\Big),$$

for $h\in[\![0,2\gamma]\!]$.

We observe that the accuracy result by Proposition 2.3.1 can be linked to the theory of wavelets in Section 2.1. This shall be discussed in Section 2.3.2.

The multidimensional extension for $d>1$ is conducted following the path of [Bihari and Harten, 1997] in a tensor product fashion. It is useful to present it because it provides us with a formalism to conduct the analysis in Chapter 3 for $d>1$. Let us consider $d=2$: one considers local reconstruction polynomials centered around $C_{\ell,\boldsymbol{k}}$—called $\pi_{\ell,\boldsymbol{k}}(\boldsymbol{x})$—made up only by terms belonging to the product of one-dimensional local reconstruction polynomials $\pi_{\ell,k_1}(x_1)$ and $\pi_{\ell,k_2}(x_2)$. This is

$$\pi_{\ell,\boldsymbol{k}}(\boldsymbol{x})=\sum_{m_1=0}^{2\gamma}\sum_{m_2=0}^{2\gamma}\pi^{m_1,m_2}_{\ell,\boldsymbol{k}}\Big(\frac{x_1-\boldsymbol{x}_{\ell,k}\cdot\boldsymbol{e}_1}{\Delta x_\ell}\Big)^{m_1}\Big(\frac{x_2-\boldsymbol{x}_{\ell,k}\cdot\boldsymbol{e}_2}{\Delta x_\ell}\Big)^{m_2}.$$

It can be easily shown that—thanks to the tensor product structure—the system corresponding to (2.20) becomes

$$(\boldsymbol{R}\otimes\boldsymbol{R})(\pi^{m_1,m_2}_{\ell,\boldsymbol{k}})_{m_1,m_2\in[\![0,2\gamma]\!]}=(\bar{f}_{\ell,\boldsymbol{k}+\boldsymbol{\delta}})_{\boldsymbol{\delta}\in[\![-\gamma,\gamma]\!]^2},$$

with $\otimes$ indicating the Kronecker product between matrices [Graham, 1981, Chapter 2]. A useful property of the Kronecker product is that $(\boldsymbol{R}\otimes\boldsymbol{R})^{-1}=\boldsymbol{R}^{-1}\otimes\boldsymbol{R}^{-1}$. Therefore we can inverse the tensor product by inverting each of its terms, obtaining, as for $d=1$, the prediction (2.17).

2.3.2 Details and multiresolution transform

Intuitively, the more the predicted value is close the actual value on the considered cell, the more we can assume that the underlying function locally behaves like a polynomial of degree at most 2γ, *cf.* Proposition 2.3.1. Quite the opposite, if the function or its derivatives have abrupt spatial changes, the prediction operator will be less suitable to correctly reconstruct the averages. This is precisely what is encoded in the details, which are a metric to quantify the information loss introduced by the projection operator and which carry essential information on the local regularity of the solution. Thus, they let us identify areas of the computation domain where the spatial resolution can be reduced—hence the mesh compressed—without affecting the quality of the stored solution.

Definition 2.3.3: Details

The detail $\overline{\mathsf{d}}_{\ell,\boldsymbol{k}}$ on the cell $C_{\ell,\boldsymbol{k}}$ is the difference between the actual value of the average and the predicted value by $\boldsymbol{P}_\triangle$ on this cell, that is

$$\overline{\mathsf{d}}_{\ell,\boldsymbol{k}} := \overline{\mathsf{f}}_{\ell,\boldsymbol{k}} - \hat{\overline{\mathsf{f}}}_{\ell,\boldsymbol{k}}, \tag{2.23}$$

for $\ell \in [\![\underline{\ell}+1, \overline{\ell}]\!]$ and $\boldsymbol{k} \in [\![0, N_\ell[\![^d$.

Otherwise said, the detail is the difference between the actual average on a cell and the predicted value obtained by the prediction operator fed with the result of the projection operator, *cf.* [Duarte, 2011, Equation (3.72)]. Details on cells having the same parent (siblings) are redundant as a consequence of the consistency property (2.15) in Definition 2.3.2. Hence, the following linear constraint holds

$$\sum_{\boldsymbol{\delta}\in\Sigma} \overline{\mathsf{d}}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}} = 0, \qquad \ell \in [\![\underline{\ell}, \overline{\ell}[\![, \quad \boldsymbol{k} \in [\![0, N_\ell[\![^d. \tag{2.24}$$

For this reason, when $d = 1$, only one detail for two siblings is significant, because $\overline{\mathsf{d}}_{\ell+1,2k} = -\overline{\mathsf{d}}_{\ell+1,2k+1}$. When $d = 2$, only three out of four details are significant. When $d = 3$, only seven out of eight details need to be considered. This calls for the introduction of the set of significant details at each level $\nabla_\ell \subset \{(\ell,\boldsymbol{k}) : \boldsymbol{k} \in [\![0, N_\ell[\![^d\}$ for $\ell \in [\![\underline{\ell}+1, \overline{\ell}]\!]$, in which for each 2^d siblings sharing the same parent, we eliminate one of them to avoid the redundancy maintained by (2.24). To provide an example, when $d = 1$, we decide to keep only the even (left) sibling, yielding

$$\nabla_\ell = \{(\ell,k) : k \in [\![0, N_\ell[\![\quad \text{and} \quad k \text{ even}\} \qquad \text{for} \quad \ell \in [\![\underline{\ell}+1, \overline{\ell}]\!].$$

Since the details are not defined for the minimum level $\underline{\ell}$, we keep all the cells considering $\nabla_{\underline{\ell}} := \{(\underline{\ell},\boldsymbol{k}) : \boldsymbol{k} \in [\![0, N_{\underline{\ell}}[\![^d\}$. In this way, we have a one-to-one correspondence between data discretized on the finest level $\overline{\ell}$ and the data at the coarsest level $\underline{\ell}$ plus the details at each level $\ell \in [\![\underline{\ell}+1, \overline{\ell}]\!]$ (*cf.* Section 2.1), upon removing the redundancy (2.24):

$$\overline{\mathbf{f}}_{\overline{\ell}} \quad \underset{\mathcal{M}_{\mathcal{R}}^{-1}}{\overset{\mathcal{M}_{\mathcal{R}}}{\rightleftarrows}} \quad (\overline{\mathbf{f}}_{\underline{\ell}}, \overline{\mathbf{d}}_{\underline{\ell}+1}, \dots, \overline{\mathbf{d}}_{\overline{\ell}}), \tag{2.25}$$

where $\overline{\mathbf{f}}_\ell = (\overline{\mathsf{f}}_{\ell,\boldsymbol{k}})_{\boldsymbol{k}\in[\![0,N_\ell[\![^d}$ for $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ and $\overline{\mathbf{d}}_\ell = (\overline{\mathsf{d}}_{\ell,\boldsymbol{k}})_{(\ell,\boldsymbol{k})\in\nabla_\ell}$ for $\ell \in [\![\underline{\ell}+1, \overline{\ell}]\!]$. Ultimately, $\mathcal{M}_{\mathcal{R}}$ is nothing but a change of basis which, by yielding the details, emphasized certain features of the data in terms of local smoothness. Indeed, each side of (2.25) contains the same number of elements:

$$\#\big(\overline{\mathbf{f}}_{\overline{\ell}}\big) = N_{\overline{\ell}} = 2^{d\overline{\ell}},$$

$$\#\big((\overline{\mathbf{f}}_{\underline{\ell}}, \overline{\mathbf{d}}_{\underline{\ell}+1}, \dots, \overline{\mathbf{d}}_{\overline{\ell}})\big) = N_{\underline{\ell}} + \sum_{\ell=\underline{\ell}+1}^{\overline{\ell}} (2^{d\ell} - 2^{d(\ell-1)}) = N_{\underline{\ell}} + N_{\overline{\ell}} - N_{\underline{\ell}} = N_{\overline{\ell}}.$$

The previous multiresolution analysis can be linked [Hovhannisyan and Müller, 2010, Theorem 2.1] to the wavelet theory as follows, see also [Müller, 2002, Chapter 2], [Cohen et al., 2003] and [Duarte, 2011, Part 1]. The averages $\overline{\mathsf{f}}$ of an underlying function f are defined using the "dual scaling function" $\tilde{\phi}_{\ell,\boldsymbol{k}} := \mathbb{1}_{C_{\ell,\boldsymbol{k}}}/|C_{\ell,\boldsymbol{k}}|_d$ by taking the duality product $\overline{\mathsf{f}}_{\ell,\boldsymbol{k}} = \langle f, \tilde{\phi}_{\ell,\boldsymbol{k}}\rangle$. Observe that the dual scaling function is normalized in L^1 [Cohen et al., 2003], thus the conjugate space is L^∞, contrarily to the L^2 setting for orthogonal wavelets, see Section 2.1. Since we

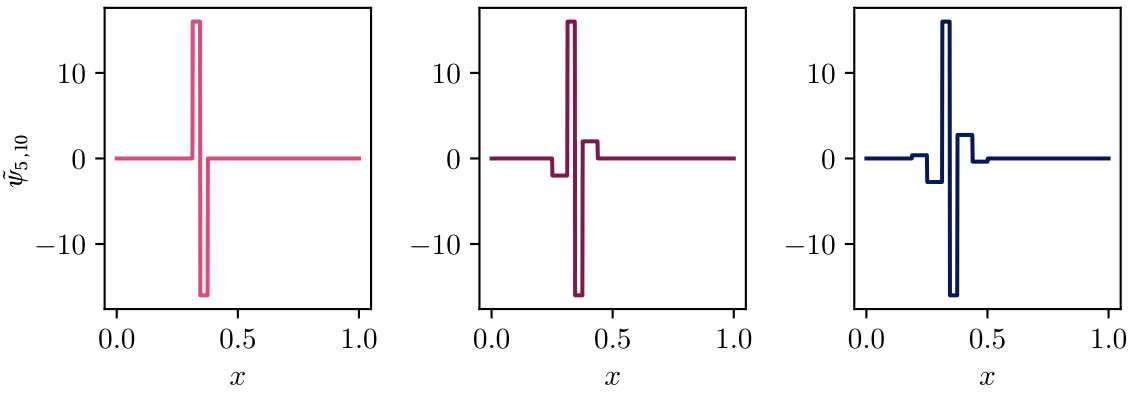


Figure 2.4: Example of dual wavelet $\tilde{\psi}_{5,10}$ for $d = 1$ using, from left to right, $\gamma = 0, 1$ and 2.

have selected linear prediction operators, see (2.16), (2.17) and (2.18), we can claim that $\hat{\bar{f}}_{\ell,\boldsymbol{k}} = \sum_{\boldsymbol{\sigma}} \psi_{\boldsymbol{\sigma}} \bar{f}_{\ell-1,\lfloor \boldsymbol{k}/2 \rfloor + \boldsymbol{\sigma}}$ for suitable weights which stencil and number depend on γ. Hence by Definition 2.3.3

$$\bar{d}_{\ell,\boldsymbol{k}} = \langle f, \tilde{\phi}_{\ell,\boldsymbol{k}} \rangle - \sum_{\boldsymbol{\sigma}} \psi_{\boldsymbol{\sigma}} \langle f, \tilde{\phi}_{\ell-1,\lfloor \boldsymbol{k}/2 \rfloor + \boldsymbol{\sigma}} \rangle = \langle f, \tilde{\psi}_{\ell,\boldsymbol{k}} \rangle,$$

where $\tilde{\psi}_{\ell,\boldsymbol{k}}$ is the "dual wavelet" being therefore given by

$$\tilde{\psi}_{\ell,\boldsymbol{k}} = \tilde{\phi}_{\ell,\boldsymbol{k}} - \sum_{\boldsymbol{\sigma}} \psi_{\boldsymbol{\sigma}} \tilde{\phi}_{\ell-1,\lfloor \boldsymbol{k}/2 \rfloor + \boldsymbol{\sigma}}.$$

For example, when $d = 1$, thus (2.16) is used, the dual wavelet is given by

$$\tilde{\psi}_{\ell,k} = \tilde{\phi}_{\ell,k} - \tilde{\phi}_{\ell-1,\lfloor k/2 \rfloor} - \sum_{\sigma=1}^{\gamma} \psi_{\sigma} (\tilde{\phi}_{\ell-1,\lfloor k/2 \rfloor + \sigma} - \tilde{\phi}_{\ell-1,\lfloor k/2 \rfloor - \sigma}),$$

where the weights are given in Table 2.1. An example of dual wavelets is given in Figure 2.4. The exactness of the prediction operator claimed in Proposition 2.3.1 can be now be reinterpreted in terms of vanishing moments for the dual wavelet. Indeed, if f is a polynomial of degree at most 2γ then $\langle f, \tilde{\psi}_{\ell,\boldsymbol{k}} \rangle = 0$, meaning that the corresponding detail is zero. Otherwise said, the dual wavelet has $2\gamma + 1$ vanishing moments. We then see that more accurate predictions, *i.e.* which dual wavelets have many vanishing moments, require large stencils and are thus more costly and difficult to implement.

The fact of changing the basis under which data are stored has the following advantage. Details are a local regularity indicator of the encoded function, as stated by the following result, see [Müller, 2002, Corollary 2, Section 2.5.3] and [Cohen et al., 2003, Equation (29)].

Proposition 2.3.2: Decay of the details

Assume that, for some $\ell \in [\![\underline{\ell}+1, \overline{\ell}]\!]$ and $\boldsymbol{k} \in [\![0, N_\ell[\![^d$, the function $f \in W^{\nu,\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))$ for some smoothness $\nu \geq 0$. Then, the following decay estimate for the details holds

$$|\bar{d}_{\ell,\boldsymbol{k}}| \lesssim 2^{-\ell \min(\nu, 2\gamma+1)} |f|_{W^{\min(\nu,2\gamma+1),\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))}, \tag{2.26}$$

where the constant depends only on γ, the width of the prediction stencil.

Proof. Consider that $\min(\nu, 2\gamma + 1) = 2\gamma + 1$ without loss of generality. We recall that both the dual scaling function $\tilde{\phi}_{\ell,\boldsymbol{k}}$ and the dual wavelet $\tilde{\psi}_{\ell,\boldsymbol{k}}$ are L^1-normalized, hence the conjugate exponent is $p = \infty$. Since $f \in W^{2\gamma+1,\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))$, [DeVore and Sharpley, 1984, Theorem 3.4] shows that there exists a polynomial π of degree at most 2γ such that $\|f - \pi\|_{L^\infty(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))} \lesssim |\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}})|^{(2\gamma+1)/d} |f|_{W^{2\gamma+1,\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))}$, where the constant

hidden behind the notation $\lesssim$ depends only on γ. We therefore have

$$|\overline{\mathrm{d}}_{\ell,\boldsymbol{k}}| = |\langle f, \tilde{\psi}_{\ell,\boldsymbol{k}}\rangle| = |\langle f-\pi, \tilde{\psi}_{\ell,\boldsymbol{k}}\rangle + \langle \pi, \tilde{\psi}_{\ell,\boldsymbol{k}}\rangle| = |\langle f-\pi, \tilde{\psi}_{\ell,\boldsymbol{k}}\rangle| \le \|f-\pi\|_{L^\infty(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))}\|\tilde{\psi}_{\ell,\boldsymbol{k}}\|_{L^1}$$
$$\lesssim |\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}})|^{(2\gamma+1)/d}|f|_{W^{2\gamma+1,\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))} \le 2^{-\ell(2\gamma+1)}|f|_{W^{2\gamma+1,\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))},$$

where we use the property on the vanishing moments of the dual wavelet, *cf.* Proposition 2.3.1, the Hölder inequality, the property of π, and the fact that $|\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}})| \lesssim 2^{-\ell d}$ (again, the constant depends only on γ, thus choice of prediction operator and thus of dual wavelet). □

Remark 2.3.2. *Observing that in the considered case, the dual scaling function $\tilde{\phi}_{\ell,\boldsymbol{k}}$ and the dual wavelet $\tilde{\psi}_{\ell,\boldsymbol{k}}$ are also normalized in every L^q norm, with different constants, [DeVore and Sharpley, 1984, Theorem 3.4] also guarantees weaker bounds on the details like*

$$|\overline{\mathrm{d}}_{\ell,\boldsymbol{k}}| \lesssim 2^{-\ell\min(\nu,2\gamma+1)}|f|_{W^{\min(\nu,2\gamma+1),p}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))},$$

for $f \in W^{\nu,p}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))$ for $p \in [\![1,+\infty]\!]$. Observe that similar estimations can be proposed using the more involved Besov spaces instead of Sobolev spaces, see [Jaming and Malinnikova, 2016] and references therein.

The estimate (2.26) means that the details decrease with larger level ℓ if the regularity ν is such that $\nu > 0$ (namely if the solution is more than bounded), according to the smoothness of the function. In this case, for smooth function (*i.e.* when $\nu \ge 2\gamma+1$), it can be useful to increase γ to achieve a faster decay of the details. Quite the opposite, close to a jump discontinuity $\nu = 0$—the typical situation with problems involving shocks and Riemann problems [LeVeque, 2002, Chapter 1], [Godlewski and Raviart, 2013, Chapter 1]—details have constant magnitude throughout the levels. In this latter case, using high-order but expensive predictions with large γ is not particularly useful, since this will not accelerate the decay of the details according to (2.26).

Remark 2.3.3 (Density functions *versus* moments)**.** *Since the Sobolev spaces $W^{\nu,\infty}$ are algebras, we can infer the regularity of the distribution densities $\overline{\mathbf{f}}$ from the expected regularity of the moments $\overline{\mathbf{m}}$ (obtained by the application of the matrix $\boldsymbol{M}$), in particular the conserved ones. This is important because the conserved moments are eventually the quantities we are interested in and which have corresponding continuous equations under the form of conservation laws.*

Looking at the decay estimate (2.26), we see that the multiresolution decomposition describes the local regularity of the function essentially as the Fourier series describes the global smoothness of a periodic signal. Indeed, let $f \in C^m([0,1])$ for some $m \in \mathbb{N}$ and periodic, then [Stein and Shakarchi, 2011, Chapter 3]

$$\left|\int_0^1 e^{-2i\pi nx} f(x)\mathrm{d}x\right| = o\left(\frac{1}{|n|^m}\right), \qquad n \in \mathbb{Z}.$$

where the role of the detail is taken by the amplitude of each mode in the decomposition and the number of the harmonics n plays the role of 2^ℓ. For multiresolution, the detail $\overline{\mathrm{d}}_{\ell,k}$ measures the frequential content of the encoded function at frequency 2^ℓ around the point $2^{-\ell}k$.

In order numerically check that (2.26) is indeed sharp and can be used to predict the magnitude of details—even when they are not available, *cf.* Section 2.4.2 —with good fidelity, we consider the case $d = 1$, $\gamma = 1$ on a domain $\Omega = [-3,3]$. The following functions with different smoothness are considered

$$\text{(a)}\quad f(x) = e^{-20x^2} \in W^{\infty,\infty}(\Omega), \qquad \text{(b)}\quad f(x) = (1+x)\mathbb{1}_{[-1,0]}(x) + (1-x)\mathbb{1}_{[0,1]}(x) \in W^{1,\infty}(\Omega), \tag{2.27}$$
$$\text{(c)}\quad f(x) = \sqrt{x}\mathbb{1}_{[0,1]}(x) + \frac{3-x}{2}\mathbb{1}_{[1,3]}(x) \in W^{1/2,\infty}(\Omega), \qquad \text{(d)}\quad f(x) = \frac{1+x}{2}\mathbb{1}_{[-1,1]}(x) \in W^{0,\infty}(\Omega).$$

We utilize the multiresolution transform and we monitor $\max_k|\overline{\mathrm{d}}_{\ell,k}|$ relative to the cell where the homogeneous Sobolev norm is attained, thus maximal. We also study the ratio $\max_k|\overline{\mathrm{d}}_{\ell,k}|/\max_k|\overline{\mathrm{d}}_{\ell+1,k}|$. We obtain what is presented in Table 2.2, showing a very fine agreement with (2.26), meaning that we correctly recover $\max_k|\overline{\mathrm{d}}_{\ell,k}|/\max_k|\overline{\mathrm{d}}_{\ell+1,k}| = 2^{\min(\nu,2\gamma+1)}$, where ν is the regularity of the datum. We remark that for the most regular function, the size of the details is limited by the choice of prediction operator ($2\gamma+1$ in this case), whereas

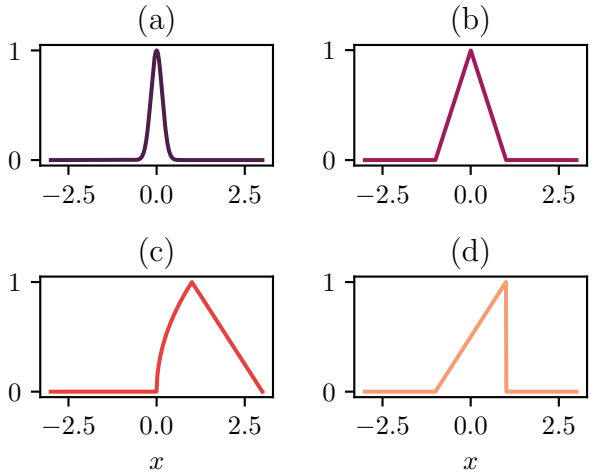


Figure 2.5: Functions (2.27) to test the decay estimates for the details

Table 2.2: Empirical detail decay, measuring the maximum detail.

ℓ	(a)		(b)		(c)		(d)	
	$\max_k \lvert\bar{d}_{\ell,k}\rvert$	$\frac{\max_k \lvert\bar{d}_{\ell,k}\rvert}{\max_k \lvert\bar{d}_{\ell+1,k}\rvert}$	$\max_k \lvert\bar{d}_{\ell,k}\rvert$	$\frac{\max_k \lvert\bar{d}_{\ell,k}\rvert}{\max_k \lvert\bar{d}_{\ell+1,k}\rvert}$	$\max_k \lvert\bar{d}_{\ell,k}\rvert$	$\frac{\max_k \lvert\bar{d}_{\ell,k}\rvert}{\max_k \lvert\bar{d}_{\ell+1,k}\rvert}$	$\max_k \lvert\bar{d}_{\ell,k}\rvert$	$\frac{\max_k \lvert\bar{d}_{\ell,k}\rvert}{\max_k \lvert\bar{d}_{\ell+1,k}\rvert}$
16	4.65e-13	–	3.81e-6	–	4.72e-4	–	1.25e-1	–
15	3.72e-12	8.00	7.63e-6	2.00	6.57e-4	1.39	1.25e-1	1.00
14	2.98e-11	8.00	1.53e-5	2.00	9.23e-4	1.41	1.25e-1	1.00
13	2.38e-10	8.00	3.05e-5	2.00	1.30e-3	1.41	1.25e-1	1.00
12	1.91e-9	8.00	6.10e-5	2.00	1.84e-3	1.41	1.25e-1	1.00
11	1.52e-8	8.00	1.22e-4	2.00	2.60e-3	1.41	1.25e-1	1.00
10	1.22e-7	8.00	2.44e-4	2.00	3.68e-3	1.41	1.25e-1	1.00
9	9.75e-7	8.00	4.88e-4	2.00	5.21e-3	1.41	1.25e-1	1.00
8	7.79e-6	7.99	9.77e-4	2.00	7.37e-3	1.41	1.25e-1	1.00
7	6.22e-5	7.99	1.95e-3	2.00	1.04e-2	1.41	1.25e-1	1.00
6	4.90e-4	7.88	3.91e-3	2.00	1.47e-2	1.41	1.26e-1	1.00
5	3.60e-3	7.35	7.81e-3	2.00	2.08e-2	1.41	1.27e-1	1.01
4	1.96e-2	5.43	1.56e-2	2.00	2.95e-2	1.41	1.29e-1	1.02
3	1.26e-1	6.43	3.13e-2	2.00	4.17e-2	1.41	1.33e-1	1.03
	Theoretical	8		2		$\sqrt{2}$		1

for less regular choices, it is the regularity of the function which determines the decay ratio (when $\nu = 1, 1/2$ and 0, for (b), (c) and (d)). This confirm the reliability of (2.26), that we shall employ in Section 2.4.2.

2.3.3 Trees, grading and reconstruction operator

We define the notion of tree and the property of grading of such tree, which allows to implement the multiresolution transform in an optimal way [Cohen et al., 2003, Proposition 2.3]. Even if this is the theoretical background in which the method lies, we shall never explicitly construct the tree structures in order to avoid their intrinsic recursive nature. We introduce the set of all significant indices given by

$$\nabla := \bigcup_{\ell=\underline{\ell}}^{\bar{\ell}} \nabla_\ell .$$

In order to guarantee the feasibility of all the operations involved with the multiresolution and because it naturally provides a multilevel covering of the domain Ω, see Figure 2.6, we want that our structure represents a tree according to the following definition.

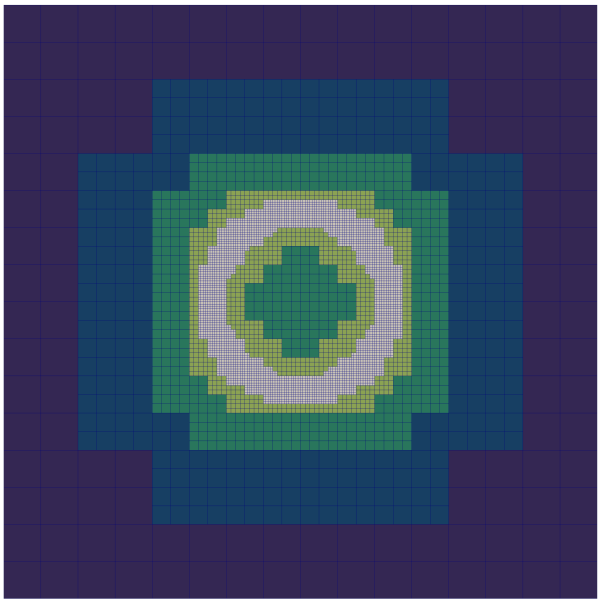


Figure 2.6: Example for $d = 2$ of multilevel covering of the domain Ω which can be interpreted in terms of trees. Different colors represent different levels of resolution spanning $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$. In particular, $\underline{\ell} = 4$ is in dark blue and $\overline{\ell} = 8$ is in white.

Definition 2.3.4: Tree

A set of indices $\Lambda \subset \nabla$ represents a tree if

- $\nabla_{\underline{\ell}} \subset \Lambda$, *i.e.* the coarsest level wholly belongs to the structure.
- If $(\ell, \boldsymbol{k}) \in \Lambda$, then its $2^d - 2$ siblings belong to Λ as well.
- If $(\ell, \boldsymbol{k}) \in \Lambda$ is such that its $2^d - 1$ children are in Λ, then its parent $(\ell - 1, \lfloor \boldsymbol{k}/2 \rfloor)$ has the same property as well.

In the case $d = 1$, Definition 2.3.4 simplifies—because $2^d - 2 = 0$—and just requests that

- $\nabla_{\underline{\ell}} \subset \Lambda$.
- If $(\ell, k) \in \Lambda$, then $(\ell - 1, \lfloor k/2 \rfloor) \in \Lambda$, namely no orphan cell exists in the structure.

Given a tree $\Lambda \subset \nabla$ satisfying Definition 2.3.4, since we have discarded some cells in ∇ to avoid detail redundancy (2.24), we indicate $R(\Lambda)$ the "complete tree". This is the set of elements in Λ (detail cells) completed by their siblings missed due to the construction of ∇_ℓ (non-detail cells). If in the case $d = 1$, we choose to consider only the even (left) cells as detail cells, thus we obtain

$$R(\Lambda) = \nabla_{\underline{\ell}} \cup \{(\ell, k), (\ell, k+1) \ : \ (\ell, k) \in \Lambda, \text{ for } \ell \in [\![\underline{\ell} + 1, \overline{\ell}]\!]\}.$$

Notice that $\Lambda \subsetneq R(\Lambda) \not\subset \nabla$. We also introduce the set of leaves $L(\Lambda) \subset \Lambda$, which are the elements of Λ without child. Adding the non-detail cells to $L(\Lambda)$ in the usual fashion, we obtain the complete leaves $S(\Lambda)$, which are depicted in Figure 2.6. We have $L(\Lambda) \subsetneq S(\Lambda) \not\subset \nabla$. Again, for $d = 1$, we have

$$S(\Lambda) = \{(\underline{\ell}, k) \in L(\Lambda)\} \cup \{(\ell, k), (\ell, k+1) \ : \ (\ell, k) \in L(\Lambda) \text{ with } \ell \in [\![\underline{\ell} + 1, \overline{\ell}]\!]\}.$$

As observed by [Cohen et al., 2003], the cells $(C_{\ell,\boldsymbol{k}})_{(\ell,\boldsymbol{k}) \in S(\Lambda)}$ form a hybrid partition of the domain Ω, meaning that they are all pairwise disjoint and

$$\bigcup_{(\ell,\boldsymbol{k}) \in S(\Lambda)} C_{\ell,\boldsymbol{k}} = \Omega.$$

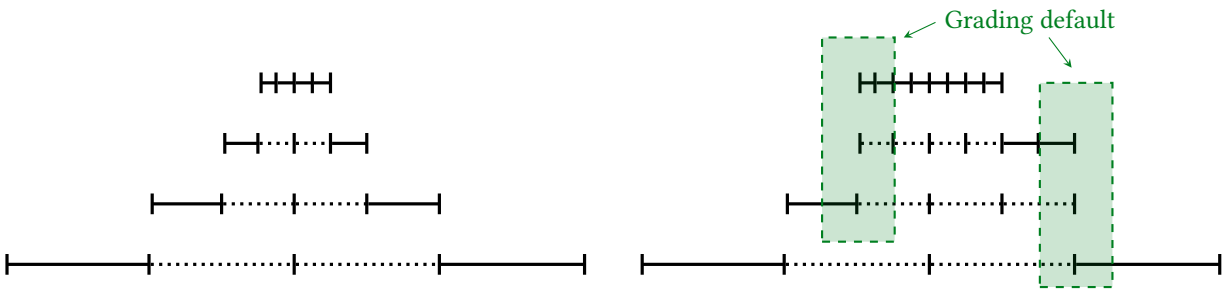


Figure 2.7: Graded tree (left) *versus* non-graded tree (right) in the case $d = 1$ for $\gamma = 1$. The complete leaves $S(\Lambda)$ are marked with full lines whereas the remaining cells $R(\Lambda) \setminus S(\Lambda)$ of the complete tree are rendered with dotted lines.

Furthermore, [Cohen et al., 2003, Proposition 2.3] have shown that one can perform the multiresolution transform in an optimal way if the resulting tree structure Λ is graded with respect to the stencil γ of the prediction operator $\boldsymbol{P}_\triangle$ (see Figure 2.7 and explication below). Still, observe that the lack of grading does not prevent one from performing the multiresolution analysis [Cohen et al., 2003, Remark 2.4], because the notion of detail (*cf.* Definition 2.3.3) and the decay estimates (*cf.* Proposition 2.3.2) are still available.

Definition 2.3.5: Grading

Let $\Lambda \subset \nabla$ be a tree according to Definition 2.3.4. Then, the tree Λ is said to be graded with respect to the prediction operator $\boldsymbol{P}_\triangle$ if for every cell in $R(\Lambda) \setminus \nabla_{\underline{\ell}}$, the cells in its prediction stencil also belong to $R(\Lambda)$. Considering the prediction operators by (2.16), (2.17) and (2.18), this is equivalent to

$$\text{If} \quad (\ell, \boldsymbol{k}) \in R(\Lambda) \setminus \nabla_{\underline{\ell}}, \quad \text{then} \quad (\ell - 1, \lfloor \boldsymbol{k}/2 \rfloor + \boldsymbol{\delta}) \in R(\Lambda), \quad \text{for} \quad \boldsymbol{\delta} \in [\![-\gamma, \gamma]\!]^d.$$

In the sequel, given a tree structure Λ according to Definition 2.3.4, the operation yielding the smallest graded tree containing Λ shall be indicated by $\mathcal{G}(\Lambda)$. The grading property is important because it guarantees that we can implement the isomorphism between

$$(\bar{\mathsf{f}}_{\ell,\boldsymbol{k}})_{(\ell,\boldsymbol{k}) \in S(\Lambda)} \quad \longleftrightarrow \quad (\bar{\mathbf{f}}_{\underline{\ell}}, (\bar{\mathsf{d}}_{\ell,\boldsymbol{k}})_{(\ell,\boldsymbol{k}) \in \Lambda \setminus \nabla_{\underline{\ell}}}), \tag{2.28}$$

in an efficient manner. This means that it is equivalent to know averages on the complete leaves $S(\Lambda)$ of a graded tree Λ or to have averages on $\nabla_{\underline{\ell}}$ at the coarsest level $\underline{\ell}$ plus the details of $\Lambda \setminus \nabla_{\underline{\ell}}$ at levels $\ell \in [\![\underline{\ell} + 1, \overline{\ell}]\!]$ available. In this work, we choose to store information on the complete leaves $S(\Lambda)$.

Let now $\Lambda \subset \nabla$ be a graded tree and assume to know the averages on the complete leaves $(\bar{\mathsf{f}}_{\ell,\boldsymbol{k}})_{(\ell,\boldsymbol{k}) \in S(\Lambda)}$. From this information, we can build the reconstruction on all cells at the finest level $\overline{\ell}$, which shall be paramount to construct adaptive numerical schemes:

$$\hat{\hat{\bar{\mathbf{f}}}}_{\overline{\ell}} = (\hat{\hat{\bar{\mathsf{f}}}}_{\overline{\ell},\boldsymbol{k}})_{\boldsymbol{k} \in [\![0, N_{\overline{\ell}}[\![^d}, \tag{2.29}$$

where the double hat represents the reconstruction operator. With this operator, the information, stored on the complete leaves $S(\Lambda)$, is propagated from coarse (at the local level of resolution of $S(\Lambda)$) to the finest level $\overline{\ell}$ by means of level-by-level applications of the prediction operator $\boldsymbol{P}_\triangle$, without adding the (unavailable) details. The reconstruction operator yields reconstructions of the lacking information on (possibly) virtual cells at the finest level using the values stored on the complete leaves $S(\Lambda)$ at the local level of refinement.

2.3.4 Mesh thresholding

The passage from one representation to the other illustrated on (2.28) is performed by the so-called "fast wavelet transform". Apart from this equivalence, the decomposition in terms of details is superior as far as we want to probe the local regularity of the functions, *cf.* Proposition 2.3.2. This can be exploited to coarsen the computational

mesh in areas where the solution is strongly regular, still being sure that we can reconstruct (2.29) information with accuracy within a certain given tolerance $0 < \epsilon \ll 1$. This is done, theoretically, by setting to zero the details which are below a certain value. From the practical point of view, one really eliminates the corresponding cells from the data structure at the end of the process.

Let $\Lambda \subset \nabla$ be a graded tree and $(\bar{f}^j_{\ell,\boldsymbol{k}})_{(\ell,\boldsymbol{k})\in S(\Lambda)}$ the datum defined on its leaves, for $j \in [\![1,q]\!]$ spanning all the distribution functions associated with the discrete velocities. In order to yield a tree structure when performing the mesh thresholding that we shall describe in a moment, we must treat detail cells having the same parent at once. This is done by considering the same detail for all of them, see [Müller, 2002, Algorithm 4, Section 3.6]:

$$|\bar{d}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}|_\infty := \max_{\boldsymbol{\pi}\in\Sigma} |\bar{d}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\pi}}|, \tag{2.30}$$

for $\ell \in [\![\underline{\ell}, \overline{\ell}[\![$ and $\boldsymbol{k} \in [\![0, N_\ell[\![^d$. This becomes trivial when $d = 1$ since the details on two siblings have the same modulus, as previously observed. The use of a L^∞ metric for the details naturally yields control in any other L^p: we shall mainly be interested in measuring L^1 errors. Other authors [Duarte, 2011, Equation (7.15)] consider the L^2 framework, with

$$|\bar{d}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}|_2 := \Big(2^{-d} \sum_{\boldsymbol{\pi}\in\Sigma} |\bar{d}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\pi}}|^2\Big)^{1/2}.$$

Observe that our L^∞ estimation shall always be more restrictive than the L^2 one and indeed any L^p one. Then, the thresholding operator is constructed as

$$\mathcal{T}_\epsilon(\Lambda) := \nabla_{\underline{\ell}} \cup \Big\{(\ell,\boldsymbol{k}) \in \Lambda \ : \ \max_{j\in[\![1,q]\!]} |\bar{d}^j_{\ell,\boldsymbol{k}}|_\infty > \epsilon_\ell\Big\}, \tag{2.31}$$

where $\epsilon_{\underline{\ell}+1}, \dots, \epsilon_{\overline{\ell}} \geq 0$ form a sequence of level-dependent non-negative thresholds to be defined. In this way, we end up with a thresholded compressed mesh which is the same for every field spanned by $j \in [\![1,q]\!]$ (*cf.* [Müller, 2002, Equation (3.38)]) and is constructed by the most restrictive inequality on the details. We obtain [Cohen et al., 2003, Equation (43) and (44)] and [Müller, 2002, Theorem 6]

Proposition 2.3.3: Error control while thresholding

Let $\epsilon > 0$ be a threshold and consider a graded tree $\Lambda \subset \nabla$ according to Definition 2.3.5 with data known on the complete leaves $S(\Lambda)$. Consider the choice of level-wise thresholds ϵ_ℓ given by

$$\epsilon_\ell = 2^{-d\Delta\ell}\epsilon, \qquad \ell \in [\![\underline{\ell}+1, \overline{\ell}]\!], \tag{2.32}$$

where we recall that $\Delta\ell = \overline{\ell} - \ell$. Consider the truncation operator T_Λ putting the details corresponding to indices which are not in Λ to zero, and $A_\Lambda := \mathcal{M}_{\mathcal{R}}^{-1} T_\Lambda \mathcal{M}_{\mathcal{R}}$. Then for any $p \in [\![1,\infty]\!]$ and $j \in [\![1,q]\!]$

$$\big\|\hat{\bar{\mathbf{f}}}^j_{\overline{\ell}} - A_{\mathcal{G}\circ\mathcal{T}_\epsilon(\Lambda)} \hat{\bar{\mathbf{f}}}^j_{\overline{\ell}}\big\|_{\ell^p} \leq C_{\mathrm{MR}}\epsilon, \tag{2.33}$$

where the constant $C_{\mathrm{MR}} = C_{\mathrm{MR}}(\gamma, p) > 0$.

A similar estimate clearly holds when collecting all the distribution functions spanned by $j \in [\![1,q]\!]$ together and use any norm on $\mathbb{R}^q$. This result means that we can discard cells with small details still being able to reconstruct at the finest level $\overline{\ell}$ within a given precision controlled by ϵ. The control naturally holds when $p = 1$, see [Müller, 2002, Theorem 6, Section 5.2]. Moreover, it is also valid for $p = \infty$ thanks to (2.30), see [Cohen et al., 2003, (45)]. Indeed, it is also valid for any $p \in [\![1,\infty]\!]$ by interpolation [Cohen et al., 2003] as we have emphasized that our L^∞ estimation is always be more restrictive than any L^p control. The price to pay is a constant depending on p because of the different normalization constants for the wavelets according to the chosen norm. Let us comment on the choice of level-wise thresholds (2.32) in Proposition 2.3.3. The dependence on d comes from the fact that in the proof, the number of discarded detail cells which are not in Λ is bounded from above by the crude estimate $\#(\nabla) = N^d_{\overline{\ell}} = 2^{d\overline{\ell}}$. The dependency on ℓ is coherent with the detail decay estimate (2.26), stating that if the underlying function is slightly more than just bounded, the details shall decrease with ℓ. For this reason, larger

thresholds can be allowed for larger ℓ.

2.4 Dynamic mesh evolution using multiresolution

This thresholding procedure is just able to eliminate cells from the structure. Still, we also need to enlarge the mesh because the systems we aim at approximating are time-dependent, so that we have to ensure that the mesh is suitable to represent the solution at the next time step—which is unknown at the time the mesh is evolved—within a certain accuracy [Cohen et al., 2003] or [Müller, 2002, Hovhannisyan and Müller, 2010] where this feature is called "reliability".

To achieve this, we observe that our schemes (2.13) and (2.14) feature two mechanisms (also see the works on Finite Volume schemes [Müller, 2002, Section 4.1.2]), namely

- the propagation of information at finite speed *via* the stencil in the stream phase (2.14);
- and the non-linearity in the collision phase (2.13) which can yield a regularity loss even for smooth initial data.

At each discrete time $t \in \Delta t\mathbb{N}$, we possess a solution defined on the complete leaves $S(\Lambda(t))$ of the graded tree $\Lambda(t) \subset \nabla$. To compute the solution at the next time step $t + \Delta t \in \Delta t\mathbb{N}$, we need to ensure that the computational lattice is refined enough so that an upper-bound similar to (2.33) still holds for the new solution. Of course, due to the fact that at the moment of constructing the mesh, the new solution is still unknown, we have to devise a heuristics to slightly enlarge the tree $\Lambda(t)$ with the information known at time $t \in \Delta t\mathbb{N}$. The way of operating is resumed as follows

$$\Lambda(t) \quad \xrightarrow{\mathcal{T}_\epsilon} \quad \mathcal{T}_\epsilon(\Lambda(t)) \quad \xrightarrow{\mathcal{H}_\epsilon} \quad \mathcal{H}_\epsilon \circ \mathcal{T}_\epsilon(\Lambda(t)) \quad \xrightarrow{\mathcal{G}} \quad \mathcal{G} \circ \mathcal{H}_\epsilon \circ \mathcal{T}_\epsilon(\Lambda(t)) =: \Lambda(t + \Delta t), \tag{2.34}$$

where the details used by $\mathcal{H}_\epsilon$—which we still have to define—and $\mathcal{T}_\epsilon$ (2.31) are those of the old solution, namely $(\bar{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t))_{(\ell,\boldsymbol{k})\in\Lambda(t)\setminus\nabla_\ell}$ for $j \in [\![1, q]\!]$. In the previous expression, we have

- $\mathcal{T}_\epsilon$, the threshold operator (2.31), which eliminates superfluous cells. It can only merge fine cells on the tree to yield coarser ones.
- $\mathcal{H}_\epsilon$, the enlargement operator, which breaks cells to form finer ones and is constructed to slightly enlarge the structure in order to accommodate the slowly evolving solution at the new time $t + \Delta t \in \Delta t\mathbb{N}$.
- $\mathcal{G}$, is the grading operator, which can also refine cells.

The nonlinear dependency of these operators on the solution defined on $S(\Lambda(t))$ is not written explicitly for the sake of keeping notation simple.

2.4.1 Addition of neighboring cells

Since we expect propagation of information at finite speed *via* the stream phase (1.4)/(2.13) of the lattice Boltzmann method, we want to ensure that this flux of information is correctly captured by the computational mesh. Inspired by [Harten, 1994], we thus request that $\mathcal{H}_\epsilon$ does the following:

$$\text{If} \quad (\ell, \boldsymbol{k}) \in R(\mathcal{T}_\epsilon(\Lambda(t))), \quad \text{then} \quad (\ell, \boldsymbol{k} - \boldsymbol{c}_j) \in R(\mathcal{H}_\epsilon \circ \mathcal{T}_\epsilon(\Lambda(t))), \quad \text{for} \quad j \in [\![1, q]\!]. \tag{2.35}$$

This means that, at each level of refinement, we add also the neighboring cells at the same level of refinement according to the discrete velocities of the lattice Boltzmann scheme at hand. The formula stipulates that if a cell at a certain level of refinement ℓ is kept in the structure, then we also keep some of its neighbors at the same level ℓ. The number of kept neighbors is determined by the largest shift associated with the discrete velocities of the lattice Boltzmann scheme at hand. We observe that this procedure is inherent to hyperbolic equations where it has originally been developed [Harten, 1994, Harten, 1995, Cohen et al., 2003] in the realm of Finite Volume schemes. For parabolic problems, where the propagation is done at infinite speed, this procedure still guarantees

that we are able to capture all the phenomena [N'Guessan et al., 2021], because the infinite velocity is intrinsic to the continuous equations but lattice Boltzmann schemes are explicit (*e.g.* when using a D_2Q_9 with three conserved moments [Lallemand and Luo, 2000] to approximate the solution the incompressible Navier-Stokes equations), thus behave "hyperbolically" because the speed of propagation is finite.

2.4.2 Refinement based on the details

Besides adding neighbors according to the velocity stencil of the lattice Boltzmann scheme, we refine some cells based on their detail in order to identify areas where gradients can steepen in time and blowup are likely to happen. One must take into account that we need to utilize the available data to estimate what is going to happen due to the non-linearity of the collision. We propose:

$$\begin{aligned} &\text{If} \quad (\ell,\boldsymbol{k}) \in R(\mathcal{T}_\epsilon(\Lambda(t))) \quad \text{with} \quad \ell \in [\![\underline{\ell}+1, \overline{\ell}[\![\quad \text{and} \quad \max_{j\in[\![1,q]\!]} |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)|_\infty > 2^{d+\overline{\mu}}\epsilon_\ell, \\ &\text{then} \quad (\ell+1, 2\boldsymbol{k}+\boldsymbol{\delta}) \in R(\mathcal{H}_\epsilon \circ \mathcal{T}_\epsilon(\Lambda(t))) \quad \text{for} \quad \boldsymbol{\delta}\in\Sigma, \end{aligned} \tag{2.36}$$

where the parameter $\overline{\mu} \geq 0$ has to be fixed according to the expected regularity of the solution, in a way that shall be described in what follows. In this expression, the magnitude of the details between siblings is handled as in (2.30). Remark that this refinement criterion acts on 2^d siblings at once by refining all of them once the metric on the details is large enough. This refinement criterion means that if the current cell is kept because its detail is not small enough to coarsen it, but moreover the detail is quite large, then we have to refine such cell. This allows to identify areas of the mesh where the solution is undergoing a decrease of smoothness. The rationale is based on estimations such as those from Proposition 2.3.2 on details at time t to estimate those (unavailable) at time $t+\Delta t$.

The refinement criterion is devised as follows. Consider that $f^j(t+\Delta t,\boldsymbol{x})$, which averages shall be $(\overline{\mathrm{f}}^j_{\ell,\boldsymbol{k}}(t+\Delta t))_{(\ell,\boldsymbol{k})\in S(\Lambda(t+\Delta t))}$ is such that $f^j(t+\Delta t,\cdot) \in W^{\nu,\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))$ for some cell indexed by $(\ell,\boldsymbol{k}) \in S(\Lambda(t))$ and $\nu \geq 0$, indicating the local regularity of the unknown solution at the new time-step. Set $\overline{\mu} := \min(\nu, 2\gamma+1)$. Since this solution is unknown at the stage at which we are utilizing $\mathcal{H}_\epsilon$, we assume that the solution varies slowly from t to $t+\Delta t$, hence we infer

$$|\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t+\Delta t)| \simeq |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)| \simeq 2^{-\ell\overline{\mu}} |f^j(t,\cdot)|_{W^{\overline{\mu},\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))},$$

according to the detail decay estimate from Proposition 2.3.2, that we use as a sharp value according to the numerical verification of Section 2.3.2 (*cf.* Table 2.2). Let $\boldsymbol{\delta} \in \Sigma$: we find estimates for details which may not be available in the structure at time t:

$$\begin{aligned} |\overline{\mathrm{d}}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}(t+\Delta t)| \simeq |\overline{\mathrm{d}}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}(t)| &\simeq 2^{-(\ell+1)\overline{\mu}} |f^j(t,\cdot)|_{W^{\overline{\mu},\infty}(\mathrm{supp}(\tilde{\psi}_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}))} \\ &\leq 2^{-(\ell+1)\overline{\mu}} |f^j(t,\cdot)|_{W^{\overline{\mu},\infty}(\mathrm{supp}(\tilde{\psi}_{\ell,\boldsymbol{k}}))}, \end{aligned}$$

where the last inequality comes from the nesting of the lattices (2.9). Therefore, we obtain the estimation, analogous to [Duarte, 2011, Equation (3.106)]

$$|\overline{\mathrm{d}}^j_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}(t+\Delta t)| \simeq 2^{-\overline{\mu}} |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)|,$$

where something which is unknown (*i.e.* the left hand side) is estimated with something which is known (*i.e.* the right hand side) since $(\ell,\boldsymbol{k}) \in S(\Lambda(t))$. Looking at the way the truncation operator $\mathcal{T}_\epsilon(\Lambda(t+\Delta t))$ has been constructed, the cells $C_{\ell+1,2\boldsymbol{k}+\boldsymbol{\delta}}$ for $\boldsymbol{\delta}\in\Sigma$ would be kept if

$$\max_{j\in[\![1,q]\!]} |\overline{\mathrm{d}}^j_{\ell+1,2\boldsymbol{k}}(t+\Delta t)|_\infty > \epsilon_{\ell+1} = 2^d\epsilon_\ell, \quad \text{yielding} \quad \max_{j\in[\![1,q]\!]} |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)|_\infty > 2^{d+\overline{\mu}}\epsilon_\ell.$$

Since the local regularity ν of the solution at each time step is unknown, $\overline{\mu} = \min(\nu, 2\gamma+1)$ is a parameter of the simulation to be set. [Müller, 2002, (Equation 4.18)] proposes to take $\overline{\mu} = 2\gamma+1$, whereas [Harten, 1994, Equation (2.3)] suggests $\overline{\mu} = 2\gamma-1$. For the applications targeted by [Duarte, 2011, Equation (4.13)], $\overline{\mu} = -d$ is considered in

order to refine by one level at each time a non-negligible detail is found, yielding rather greedy refinement criteria. Still, in our work, we utilize changing choice for $\overline{\mu}$ according to the problem at hand, because we shall verify that this choice has an important impact on the quality of the numerical simulations. For example, if one knows that the solution of the problem has or shall develop a shock ($\nu = 0$), it is advisable to select $\overline{\mu} = 0$ in order to ensure to be able to refine a coarsened mesh if the shock is forming.

By proceeding at enlarging the computational mesh in this way by $\mathcal{H}_\epsilon$, we assume that it is suitable to represent the solution at the new time $t + \Delta t$ within a reasonable tolerance given by ϵ. This assumption is often called Harten's heuristics [Harten, 1994, Cohen et al., 2003] or reliability condition [Hovhannisyan and Müller, 2010] in the world of Finite Volume schemes and in our setting reads

Assumptions 2.4.1: Harten's heuristics

The tree $\mathcal{T}_\epsilon(\Lambda(t))$ has been enlarged into a graded tree $\Lambda(t+\Delta t) = \mathcal{G} \circ \mathcal{H}_\epsilon \circ \mathcal{T}_\epsilon(\Lambda(t))$ such that

$$\left\| \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t) - A_{\Lambda(t+\Delta t)} \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t) \right\| \leq C_{\mathrm{MR}} \epsilon, \qquad \left\| \mathbf{E} \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t) - A_{\Lambda(t+\Delta t)} \mathbf{E} \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t) \right\| \leq C_{\mathrm{MR}} \epsilon,$$

where $\|\cdot\|$ is a norm for vectors of size q derived from the $\|\cdot\|_{\ell^p}$ for the chosen $p \in [\![1, \infty]\!]$ and we recall that $\mathbf{E}$ represents the action of the reference lattice Boltzmann scheme.

Remark that since q is finite, it does not matter which norm we choose to pass from $\|\cdot\|_{\ell^p}$ for scalar solution to vectors of size q. The first inequality in Assumptions 2.4.1 is naturally fulfilled using the fact that $\mathcal{T}_\epsilon(\Lambda(t)) \subset \Lambda(t+\Delta t)$. The second inequality is potentially verified upon having enlarged the mesh using $\mathcal{H}_\epsilon$, which has been built considering how the reference scheme $\mathbf{E}$ acts on the solution. It basically means that the mesh is suitable for well representing the solution obtained by applying the reference scheme to the adaptive solution at the previous time step, reconstructed on the finest level. Observe that we do not rigorously prove that this assumption holds for our refinement strategy $\mathcal{H}_\epsilon$. As for the Finite Volume scheme, the Harten's approach that we have adopted to construct $\mathcal{H}_\epsilon$ has never proved to satisfy something like Assumptions 2.4.1 but is widely used in practice. The only achievement in terms of reliability condition has been obtained in [Cohen et al., 2003, Hovhannisyan and Müller, 2010] for Finite Volume schemes dealing with scalar conservation laws, with a quite sophisticated refinement strategy. Note that our formulation of the Harten heuristics Assumptions 2.4.1 is slightly different from the one in [Cohen et al., 2003, Hovhannisyan and Müller, 2010] because of the different order of the operations at each time step of the algorithm. However, this does not make any difference, except when dealing with the initial datum, because the order of the operations when time steps are concatenated is the same.

Once we have $\Lambda(t+\Delta t)$, we adapt the solution from $S(\Lambda(t))$ to $S(\Lambda(t+\Delta t))$. In this process, if cells are coarsened, we have to merge their data with the projection operator $\boldsymbol{P}_\triangledown$. On the other hand, when finer cells are added by $\mathcal{H}_\epsilon$ or $\mathcal{G}$, the missing information is reconstructed using the prediction operator $\boldsymbol{P}_\triangle$. We are left with the old solution at time t on the complete leaves of the new mesh $S(\Lambda(t+\Delta t))$, that is $(\bar{\mathrm{f}}^j_{\ell,\boldsymbol{k}}(t))_{(\ell,\boldsymbol{k}) \in S(\Lambda(t+\Delta t))}$ for $j \in [\![1, q]\!]$.

2.5 Lattice Boltzmann methods on adaptive grids

Now that we have described how to adapt the computational mesh dynamically in time, we have to explain how we adapt the lattice Boltzmann schemes of Section 2.2 to be utilized on these meshes. We construct three possible lattice Boltzmann methods on adaptive grids using only information stored on the complete leaves of the adapted tree $S(\Lambda(t+\Delta t))$. There are several possibilities because different ways of dealing with the collision phase (2.13) exist. Each strategy feature a different tradeoff between efficiency and accuracy. Given a cell $C_{\ell,\boldsymbol{k}}$, we consider the set $\mathcal{B}_{\ell,\boldsymbol{k}}$ of the indices of virtual cells at the finest level of refinement $\overline{\ell}$ covering $C_{\ell,\boldsymbol{k}}$, defined by

$$\mathcal{B}_{\ell,\boldsymbol{k}} := \{\boldsymbol{k} 2^{\Delta\ell} + \boldsymbol{\delta} \ : \ \boldsymbol{\delta} \in [\![0, 2^{\Delta\ell}[\![^d\}.$$

The idea is to do our best to operate "as if" the scheme were performed at the finest level $\overline{\ell}$. This ensures a behavior as close as possible to the reference scheme on the uniform mesh at finest level $\overline{\ell}$ with the possibility of estimate errors.

2.5.1 Reconstructed collision phase

We adapt the collision phase (2.13) by performing it as if we were at the finest level, using the reconstruction operator and then projecting back on the leaves. This is analogous to one of the so-called "exact source reconstruction" strategy to integrate source terms [Hovhannisyan and Müller, 2010] for Finite Volume. This strategy can be computationally expensive and is mostly of theoretical interest. We shall discuss this fact and introduce an alternative approach in what follows. Let $(\ell, \boldsymbol{k}) \in S(\Lambda(t+\Delta t))$, then for every $\overline{\boldsymbol{k}} \in \mathcal{B}_{\ell,\boldsymbol{k}}$ we perform the collision at the finest level $\overline{\ell}$ using reconstructed information:

$$\overline{\mathbf{m}}^{\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) = (\boldsymbol{I}-\boldsymbol{S})\hat{\overline{\mathbf{m}}}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) + \boldsymbol{S}\boldsymbol{m}^{\mathrm{eq}}\big(\hat{\overline{\mathrm{m}}}^{1}_{\overline{\ell},\overline{\boldsymbol{k}}}(t),\dots,\hat{\overline{\mathrm{m}}}^{N}_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\big).$$

Still, we finally aim at writing a fully adaptive scheme for the solution on the leaves $S(\Lambda(t+\Delta t))$, hence we average back using $\Delta\ell$ times the projection operator $\boldsymbol{P}_{\triangledown}$, obtaining

$$\begin{aligned}\overline{\mathbf{m}}^{\star}_{\ell,\boldsymbol{k}}(t) &= \frac{\boldsymbol{I}-\boldsymbol{S}}{2^{d\Delta\ell}}\sum_{\overline{\boldsymbol{k}}\in\mathcal{B}_{\ell,\boldsymbol{k}}}\hat{\overline{\mathbf{m}}}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) + \frac{\boldsymbol{S}}{2^{d\Delta\ell}}\sum_{\overline{\boldsymbol{k}}\in\mathcal{B}_{\ell,\boldsymbol{k}}}\boldsymbol{m}^{\mathrm{eq}}\big(\hat{\overline{\mathrm{m}}}^{1}_{\overline{\ell},\overline{\boldsymbol{k}}}(t),\dots,\hat{\overline{\mathrm{m}}}^{N}_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\big)\\ &= (\boldsymbol{I}-\boldsymbol{S})\overline{\mathbf{m}}_{\ell,\boldsymbol{k}}(t) + \frac{\boldsymbol{S}}{2^{d\Delta\ell}}\sum_{\overline{\boldsymbol{k}}\in\mathcal{B}_{\ell,\boldsymbol{k}}}\boldsymbol{m}^{\mathrm{eq}}\big(\hat{\overline{\mathrm{m}}}^{1}_{\overline{\ell},\overline{\boldsymbol{k}}}(t),\dots,\hat{\overline{\mathrm{m}}}^{N}_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\big),\end{aligned} \tag{2.37}$$

where the linear first term gives back the average on $C_{\ell,\boldsymbol{k}}$ thanks to the consistency of the prediction operator $\boldsymbol{P}_{\triangle}$, *cf.* Definition 2.3.2.

We see that the strategy is expensive because the equilibria need to be evaluated on $2^{d\overline{\ell}}$ cells at the finest level after having applied the reconstruction operator. Quite the opposite, we would like to evaluate the equilibria only $\#(S(\Lambda(t+\Delta t)))$ times as for the linear term $(\boldsymbol{I}-\boldsymbol{S})\overline{\mathbf{m}}_{\ell,\boldsymbol{k}}(t)$.

2.5.2 Leaves collision phase

For this reason, we introduce a different collision phase which equals (2.37) in the case of linear equilibria and which do not need to reconstruct everything at the finest level. The idea is to directly use the data available on the leaves $S(\Lambda(t+\Delta t))$ into the equilibria, that is, doing the approximation for every $\overline{\boldsymbol{k}} \in \mathcal{B}_{\ell,\boldsymbol{k}}$

$$\boldsymbol{m}^{\mathrm{eq}}\big(\hat{\overline{\mathrm{m}}}^{1}_{\overline{\ell},\overline{\boldsymbol{k}}}(t),\dots,\hat{\overline{\mathrm{m}}}^{N}_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\big) \simeq \boldsymbol{m}^{\mathrm{eq}}\big(\overline{\mathrm{m}}^{1}_{\ell,\boldsymbol{k}}(t),\dots,\overline{\mathrm{m}}^{N}_{\ell,\boldsymbol{k}}(t)\big).$$

This approximation is exact only if the equilibrium functions are linear. Otherwise, we can only hope to have the equality plus an error of the order of ϵ. Thus, we obtain—using the consistency of the projection operator—the collision phase

$$\overline{\mathbf{m}}^{\star}_{\ell,\boldsymbol{k}}(t) = (\boldsymbol{I}-\boldsymbol{S})\overline{\mathbf{m}}_{\ell,\boldsymbol{k}}(t) + \boldsymbol{S}\boldsymbol{m}^{\mathrm{eq}}\big(\overline{\mathrm{m}}^{1}_{\ell,\boldsymbol{k}}(t),\dots,\overline{\mathrm{m}}^{N}_{\ell,\boldsymbol{k}}(t)\big). \tag{2.38}$$

This corresponds to the so-called "naive source computation" for source terms in Finite Volume schemes [Hovhannisyan and Müller, 2010] or to the fact of using a prediction operator with $\gamma = 0$ to build up the reconstruction operator.

The collision strategy (2.38) is significantly cheaper than (2.37) because there is no need to reconstruct a piecewise constant representation of the solution on the full finest level $\overline{\ell}$. Using (2.37) would rely on the recursive nature of adaptive multiresolution and would yield an explosion of the complexity of the algorithm in most cases where a high compression rate can be reached. This holds even when memoization techniques are employed to reduce the number of evaluations due to the recursive structure of the reconstruction operator. In our case, memoization consists in caching the values predicted/reconstructed on a given cell—indexed by its integer coordinate and level—being able to directly recall them whenever needed by another computation. We verified both with 1D and 2D tests that for the problems we analyzed, the use of collision operator on the mere complete leaves (2.38) has a marginal impact on the accuracy of the adaptive method, see Section 2.8.1.3 for more details.

2.5.3 Predict-and-integrate collision phase

We introduce an "interpolated" collision technique, which is inspired by the work by [Hovhannisyan and Müller, 2010] concerning source terms in Finite Volume schemes. It aims at to reducing the computational cost of the reconstructed collision still keeping very good accuracy.

Considering a cell $C_{\ell,\boldsymbol{k}}$, one considers the local reconstruction polynomials $\pi^j_{\ell,\boldsymbol{k}}(t,\cdot)$ for the distribution function $\bar{\mathrm{f}}^j(t)$ for any $j \in [\![1,q]\!]$. The corresponding local reconstruction polynomials for the moments are constructed by $(\mu^1_{\ell,\boldsymbol{k}}(t,\cdot),\dots,\mu^q_{\ell,\boldsymbol{k}}(t,\cdot))^{\mathsf{t}} = \boldsymbol{M}(\pi^1_{\ell,\boldsymbol{k}}(t,\cdot),\dots,\pi^q_{\ell,\boldsymbol{k}}(t,\cdot))^{\mathsf{t}}$, as usual, by the moment matrix $\boldsymbol{M}$, where the local reconstruction polynomials for the distribution functions $\pi^1_{\ell,\boldsymbol{k}}(t,\cdot),\dots,\pi^q_{\ell,\boldsymbol{k}}(t,\cdot)$ are given as in (2.19). Then, we consider the following approximation in the reconstructed collision (2.37)

$$\begin{aligned}\frac{1}{2^{d\Delta\ell}}\sum_{\overline{\boldsymbol{k}}\in\mathcal{B}_{\ell,\boldsymbol{k}}} \boldsymbol{m}^{\mathrm{eq}}\big(\hat{\bar{\mathrm{m}}}^1_{\overline{\ell},\overline{\boldsymbol{k}}}(t),\dots,\hat{\bar{\mathrm{m}}}^N_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\big) &\simeq \frac{1}{|C_{\ell,\boldsymbol{k}}|_d}\int_{C_{\ell,\boldsymbol{k}}} \boldsymbol{m}^{\mathrm{eq}}\big(\mu^1_{\ell,\boldsymbol{k}}(t,\boldsymbol{x}),\dots,\mu^N_{\ell,\boldsymbol{k}}(t,\boldsymbol{x})\big)\mathrm{d}\boldsymbol{x}\\ &\simeq \frac{1}{2^{d\Delta\ell}}\sum_r w_r \boldsymbol{m}^{\mathrm{eq}}\big(\mu^1_{\ell,\boldsymbol{k}}(t,\boldsymbol{x}_r),\dots,\mu^N_{\ell,\boldsymbol{k}}(t,\boldsymbol{x}_r)\big),\end{aligned} \tag{2.39}$$

where the last approximation employs a quadrature formula with a finite number of real weights $(w_r)_r$ and with quadrature points $(\boldsymbol{x}_r)_r$. The idea of the first approximation is to replace the computation of the integral of a piecewise constant function on $C_{\ell,\boldsymbol{k}}$, where the values are obtained by reconstruction operator, by the integral of a function obtained using the local reconstruction polynomials. The procedure by (2.39) relies on the fact that the solution is expected to locally behave like a low degree polynomial, which is transformed by the equilibria into another non-linear function and hoping that the quadrature formula is accurate enough to approximate the integral over the considered cell. We call this approach "predict-and-integrate" because the local reconstruction polynomials correspond to construct the prediction operator $\boldsymbol{P}_\triangle$, *cf.* Section 2.3.1 and then we employ a quadrature formula.

2.5.4 Stream phase

Concerning the stream phase (2.14), the idea is again to reconstruct the post-collision distributions at the finest level $\overline{\ell}$ with the reconstruction operator, to stream as if we were on the finest level and then project on the leaves. Consider to work with the discrete velocity $j \in [\![1,q]\!]$. Let $(\ell,\boldsymbol{k}) \in S(\Lambda(t+\Delta t))$, then for every $\overline{\boldsymbol{k}} \in \mathcal{B}_{\ell,\boldsymbol{k}}$ we perform the stream at the finest level $\overline{\ell}$ using reconstructed information:

$$\bar{\mathrm{f}}^j_{\overline{\ell},\overline{\boldsymbol{k}}}(t+\Delta t) = \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}-\boldsymbol{c}_j}(t).$$

We then make $\Delta\ell$ applications of the projection operator $\boldsymbol{P}_\triangledown$ and use the consistency of the prediction operator $\boldsymbol{P}_\triangle$

$$\begin{aligned}\bar{\mathrm{f}}^j_{\ell,\boldsymbol{k}}(t+\Delta t) &= \frac{1}{2^{d\Delta\ell}}\sum_{\overline{\boldsymbol{k}}\in\mathcal{B}_{\ell,\boldsymbol{k}}} \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}-\boldsymbol{c}_j}(t) = \frac{1}{2^{d\Delta\ell}}\Bigg(\sum_{\overline{\boldsymbol{k}}\in\mathcal{B}_{\ell,\boldsymbol{k}}} \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) + \sum_{\overline{\boldsymbol{k}}\in\mathcal{E}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) - \sum_{\overline{\boldsymbol{k}}\in\mathcal{A}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\Bigg)\\ &= \bar{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}}(t) + \frac{1}{2^{d\Delta\ell}}\Bigg(\sum_{\overline{\boldsymbol{k}}\in\mathcal{E}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) - \sum_{\overline{\boldsymbol{k}}\in\mathcal{A}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t)\Bigg),\end{aligned} \tag{2.40}$$

where we have defined

$$\mathcal{E}^j_{\ell,\boldsymbol{k}} := (\mathcal{B}_{\ell,\boldsymbol{k}} - \boldsymbol{c}_j) \setminus \mathcal{B}_{\ell,\boldsymbol{k}}, \qquad \mathcal{A}^j_{\ell,\boldsymbol{k}} := \mathcal{B}_{\ell,\boldsymbol{k}} \setminus (\mathcal{B}_{\ell,\boldsymbol{k}} - \boldsymbol{c}_j),$$

where $\mathcal{B}_{\ell,\boldsymbol{k}} - \boldsymbol{w}$ for $\boldsymbol{w} \in \mathbb{Z}^d$ represents the element-wise subtraction of $\boldsymbol{w}$.

The cells indexed by $\mathcal{E}^j_{\ell,\boldsymbol{k}}$ render an incoming pseudo-flux in the cell $C_{\ell,\boldsymbol{k}}$, whereas those indexed by $\mathcal{A}^j_{\ell,\boldsymbol{k}}$ yield an outgoing one, see Figure 2.8. We observe that only the fluxes at the boundaries of the cell $C_{\ell,\boldsymbol{k}}$ have to be estimated using the reconstruction, which reduced the number of performed operation using mesh adaptation. More precisely, we have that $\#(\mathcal{E}^j_{\ell,\boldsymbol{k}}) = \#(\mathcal{A}^j_{\ell,\boldsymbol{k}}) \sim 2^{(d-1)\Delta\ell} \ll 2^{d\Delta\ell}$. The expressions for $\mathcal{E}^j_{\ell,k}$ and $\mathcal{A}^j_{\ell,k}$ are particularly

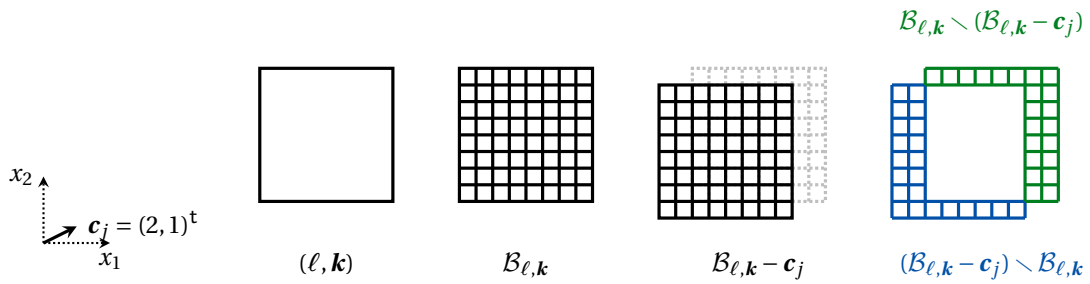


Figure 2.8: Example of the sets needed for the adaptive stream phase (2.40) for the $d = 2$ case. We consider a leaf $C_{\ell,\boldsymbol{k}}$ which in this example is at level $\ell = \overline{\ell} - 3$ and the dimensionless velocity is $\boldsymbol{c}_j = (2,1)^{\mathrm{t}}$ for illustrative purpose.

simple for $d = 1$ and read

$$\mathrm{sign}(c_j) = 0, \qquad \mathcal{E}^j_{\ell,k} = \mathcal{A}^j_{\ell,k} = \emptyset, \tag{2.41}$$

$$\mathrm{sign}(c_j) > 0, \qquad \mathcal{E}^j_{\ell,k} = \{k2^{\Delta\ell} - \delta \ : \ \delta \in [\![1, c_j]\!]\}, \quad \mathcal{A}^j_{\ell,k} = \{(k+1)2^{\Delta\ell} - \delta \ : \ \delta \in [\![1, c_j]\!]\}, \tag{2.42}$$

$$\mathrm{sign}(c_j) < 0, \qquad \mathcal{E}^j_{\ell,k} = \{(k+1)2^{\Delta\ell} - 1 + \delta \ : \ \delta \in [\![1, |c_j|]\!]\}, \quad \mathcal{A}^j_{\ell,k} = \{k2^{\Delta\ell} - 1 + \delta \ : \ \delta \in [\![1, |c_j|]\!]\}. \tag{2.43}$$

Our way of proceeding to construct the adaptive stream phase mimics the approach by [Cohen et al., 2003] performing the so-called "exact flux reconstruction" for Finite Volume schemes *via* the reconstruction operator. It can be easily shown that our approach, devised through a different way of reasoning, is indeed a CTU (Corner Transport Upwind method) Finite Volume discretization of the transport equation *cf.* (2.10) $\partial_t f^j + \boldsymbol{\xi}_j \cdot \nabla_{\boldsymbol{x}} f^j = 0$ associated with each discrete velocity, see [Colella, 1990] or [LeVeque, 2002, Chapter 20.2], with reconstruction of a piece-wise constant representation of the solution at the finest level made possible by the multiresolution analysis.

Conversely, the AMR approach from [Fakhari and Lee, 2014, Fakhari et al., 2016] relies on a Lax-Wendroff scheme for the adaptive stream phase with direct evaluation on some ghost cells at the same level of resolution ℓ. This procedure—which we shall study in Section 3.1— is surely cheaper than (2.40) but cannot ensure any control on the error because the mesh is generated by a heuristic criterion, which is not correlated to the way of locally reconstructing the solution *via* a cascade of predictions.

Remark 2.5.1. *Since the reconstruction operator utilizes $\boldsymbol{P}_\triangle$ until reaching available values stored on $S(\Lambda(t + \Delta t))$, one might use a cheaper prediction (smaller γ) to perform this operation, as hinted by [Cohen et al., 2003] with the so-called "direct evaluation"* (i.e. $\gamma = 0$)*. Such approach is used in many works, but at the cost of the error control provided by the multiresolution machinery.*

It is important to observe that once we consider a cell $C_{\ell,\boldsymbol{k}}$ adjacent to the boundary of the domain $\partial\Omega$, some virtual cell indexed by $\mathcal{E}^j_{\ell,k}$ could lie outside the domain. This calls for the enforcement of some boundary condition as explained in Section 2.7.

2.5.4.1 Non-recursive reconstruction operator: reconstruction flattening

To make computations feasible for large problems, we follow the idea of [Cohen et al., 2003, Equation (68)], claiming that in the univariate case, the recursive application of a linear prediction operator $\boldsymbol{P}_\triangle$ can be condensed into the computation of the powers of a given matrix at the beginning of the simulation, based on the assumption that the fluxes of the Finite Volume method involve only adjacent cells. For $d = 1$ and $\gamma = 1$, this reads—forgetting

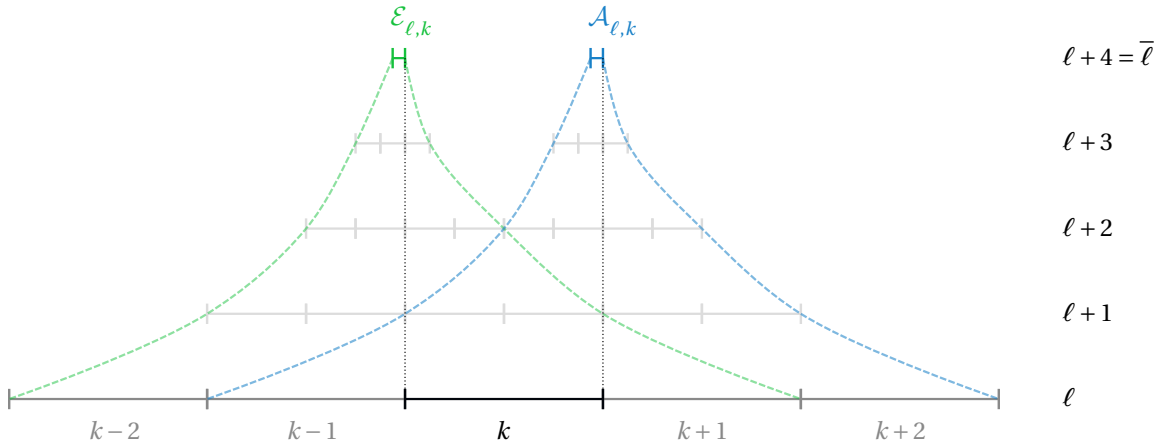


Figure 2.9: Example of non-recursive implementation of the reconstruction operator for $d=1$ and $\gamma=1$ for the velocity $+\lambda$ (dimensionless velocity $+1$). The cell on which one updates the solution is four level far from the finest level and indexed by k. The green cell on top corresponds to the cell at the finest level $\overline{\ell}$ giving the incoming pseudo-flux for the cell k at level ℓ, whereas that in blue corresponds to that of the outgoing one. The prediction operator is progressively applied spanning the intermediate (non existing) cells in pale grey inside the green (resp. blue) funnel, until reaching cells at the same level ℓ in black and grey.

about the fact that we deal with several distribution functions spanned by $j \in [\![1,q]\!]$

$$\begin{bmatrix} \hat{\hat{\mathsf{f}}}_{\overline{\ell},k2^{\Delta\ell}-2} \\ \hat{\hat{\mathsf{f}}}_{\overline{\ell},k2^{\Delta\ell}-1} \\ \hat{\hat{\mathsf{f}}}_{\overline{\ell},k2^{\Delta\ell}} \\ \hat{\hat{\mathsf{f}}}_{\overline{\ell},k2^{\Delta\ell}+1} \end{bmatrix} = \begin{bmatrix} 1/8 & 1 & -1/8 & 0 \\ -1/8 & 1 & 1/8 & 0 \\ 0 & 1/8 & 1 & -1/8 \\ 0 & -1/8 & 1 & 1/8 \end{bmatrix}^{\Delta\ell} \begin{bmatrix} \overline{\mathsf{f}}_{\ell,k-2} \\ \overline{\mathsf{f}}_{\ell,k-1} \\ \overline{\mathsf{f}}_{\ell,k} \\ \overline{\mathsf{f}}_{\ell,k+1} \end{bmatrix}, \tag{2.44}$$

for $k \in \mathbb{Z}$ and $\ell \in [\![\underline{\ell},\overline{\ell}]\!]$, where ℓ has to be thought as the local level of refinement of the mesh, *i.e.* the local level of the leaves $S(\Lambda(t+\Delta t))$. Clearly, this method works far from the boundary $\partial\Omega$ and in areas where the local refinement level of the leaves is constant. Since for $d>1$ we have constructed the prediction operator by tensor product (*cf.* Section 2.3.1.2), the matrices involved in this framework shall just be the Kronecker product $\otimes$ of that for $d=1$ in (2.44) d-times with itself.

We start to describe the procedure for general d by selecting a complete leaf $C_{\ell,\boldsymbol{k}}$ and assuming—for the moment—that it is surrounded by enough complete leaves of the same level. At the beginning of the numerical simulation, we can once for all compute by recursion (or analytically as previously described), for any level $\ell \in [\![\underline{\ell},\overline{\ell}]\!]$ and discrete velocity $j \in [\![1,q]\!]$, the set of shifts $\Xi^j_{\Delta\ell} \subset \mathbb{Z}^d$ and weights $(F^j_{\Delta\ell,\boldsymbol{\delta}})_{\boldsymbol{\delta}\in\Xi^j_{\Delta\ell}} \subset \mathbb{R}$ such that the pseudo-flux term in (2.40) is given by

$$\sum_{\overline{\boldsymbol{k}}\in\mathcal{E}^j_{\ell,\boldsymbol{k}}} \hat{\hat{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) - \sum_{\overline{\boldsymbol{k}}\in\mathcal{A}^j_{\ell,\boldsymbol{k}}} \hat{\hat{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}}(t) = \sum_{\boldsymbol{\delta}\in\Xi^j_{\Delta\ell}} F^j_{\Delta\ell,\boldsymbol{\delta}} \overline{\mathsf{f}}^{j,\star}_{\ell,\boldsymbol{k}+\boldsymbol{\delta}}(t). \tag{2.45}$$

This transforms the recursive left-hand side of (2.45) into a right-hand side made up of linear combinations of data known on the leaves with previously computed weights, in analogy with (2.44). An illustration of such a process in a one-dimensional setting is given in Figure 2.9. We might call this "recontruction flattening" because we have "flattened" the cascade of prediction spanning several levels from the current ℓ to the finest $\overline{\ell}$ solely to the level ℓ.

Quite the opposite, if the surrounding leaves at the same level are not enough (we could, for example, fall on some ghost cell), we are not sure that the value we retrieve is accurate enough according to the multiresolution analysis. Let us study this in more detail, considering $\boldsymbol{\delta} \in \Xi^j_{\Delta\ell}$. There are two possibilities:

- $C_{\ell,\boldsymbol{k}+\boldsymbol{\delta}}$ is a complete leaf belonging to $S(\Lambda(t+\Delta t))$. Everything we retrieve is adequate, because multiresolution allows us to employ this value in a recursion formula to reconstruct at finest level without adding

any detail.

- $C_{\ell,\boldsymbol{k}+\boldsymbol{\delta}}$ is not a complete leaf. There are two situations, thanks to the fact of having built a graded tree according to Definition 2.3.5. One of these two cases requires particular care.

  - It intersects a leaf at the coarser level $\ell-1$. The value we retrieve is fine because multiresolution guarantees that quantities computed by applying the prediction operator $\boldsymbol{P}_{\triangle}$ without adding details are accurate enough within the tolerance if their respective cell is situated on top of a leaf.
  - It intersects a leaf at the finer level $\ell+1$. This is the critical situation, because the retrieved value is not accurate enough to be employed in reconstructions, according to the multiresolution analysis and we have to add the proper detail information in order to preserve our target of error control.

To finish the section, let us comment on the complexity of our stream phase (2.40) using (2.45) compared to the approach by [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016] which uses a Lax-Wendroff approximation of the transport equation for each discrete velocity. We compare for $d=2$ and $\gamma=1$. This approach—valid when velocity stencils involve at most one neigboring cell in each Cartesian direction—has to recover 3 values of the solution at the considered level ℓ in the direction spanned by the discrete velocity, multiply each of them by a suitable weight and adding them up. Our approach (2.40) using (2.45) needs to recover at most 25 values—whatever the discrete velocity $\boldsymbol{c}_j$ satisfying $\max(|\boldsymbol{c}_j\cdot\boldsymbol{e}_1|,|\boldsymbol{c}_j\cdot\boldsymbol{e}_2|)\leq 2$ (spanning one or even two neighboring cells in each Cartesian direction), thus for essentially any lattice Boltzmann scheme—multiply each of them by a weight and then summing them. More generally, the number of values to recover and multiply by a weight is

$$\left(2\left\lceil\frac{\max(|\boldsymbol{c}_j\cdot\boldsymbol{e}_1|,|\boldsymbol{c}_j\cdot\boldsymbol{e}_2|)}{2}\right\rceil+1\right)^2.$$

Even if this should be quantified precisely in terms of computational cost for a given problem, a given implementation and a given architecture, the complexity of the algorithms are sensibly at the same level. The gain comes once considering that we can deal with a very large class of schemes achieving a control on the error, which is not obtained by the AMR procedures. Furthemore, our adaptive method reproduces the behavior of the reference method on the finest grid in terms of both error control and equivalent equations up to order three, instead of two for the Lax-Wendroff approach, see Section 3.1.

2.6 Error control

The major interest of adaptive meshes generated by multiresolution is that we can recover a precise error control on the perturbation (or additional) error between results on adaptive and uniform grids at the finest level $\overline{\ell}$. In this way, the perturbation introduced by adapting the grids can be mastered, in particular, *via* the threshold ϵ.

Fixing a given ℓ^p norm for $p\in[\![1,\infty]\!]$, we aim at controlling the additional error $\|\overline{\mathbf{f}}^{\mathrm{ref}}(t)-\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)\|$, where $\overline{\mathbf{f}}^{\mathrm{ref}}(t)$ is the reference solution given by $\overline{\mathbf{f}}^{\mathrm{ref}}(t+\Delta t)=\mathbf{E}\overline{\mathbf{f}}^{\mathrm{ref}}(t)$ and wholly defined at the finest level $\overline{\ell}$ and $\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)$ is the solution of the adaptive method on the adaptive mesh which have been reconstructed at the finest level, see (2.29). Here, $\|\cdot\|$ the extension of the ℓ^p norm to vectors in $\mathbb{R}^q$. Both computations start from the same initial datum on the finest grid, that is, $\Lambda(0)=\nabla$.

2.6.1 Assumptions

We introduce the natural assumptions to prove results concerning error control. They are the Harten's heuristics (Assumptions 2.4.1) and the continuity of the reference scheme. These assumptions are essentially the same than the ones for Finite Volume schemes [Cohen et al., 2003, Hovhannisyan and Müller, 2010].

Assumptions 2.6.1: Continuity of the reference scheme

The reference scheme $\mathbf{E}$ is such that there exists a constant $C_{\mathbf{E}} = 1 + \tilde{C}_{\mathbf{E}}$ with $\tilde{C}_{\mathbf{E}} \geq 0$ such that

$$\|\mathbf{E}\overline{\mathbf{u}} - \mathbf{E}\overline{\mathbf{v}}\| \leq C_{\mathbf{E}}\|\overline{\mathbf{u}} - \overline{\mathbf{v}}\|, \qquad \forall \overline{\mathbf{u}}, \overline{\mathbf{v}} \in \mathbb{R}^{qN_{\overline{\ell}}},$$

for the considered norm $\|\cdot\|$.

Remark 2.6.1. *The following procedure can be easily adapted to the context where the continuity of the scheme is measured using an ℓ^2-weighted norm as by [Junk and Yong, 2009]. It is sufficient to consider $p = 2$ and to observe that the corresponding norm (measuring the properties pertaining to the multiresolution) can be bounded by the ℓ^2-weighted norm.*

2.6.2 Error bounds and their proof

The essential idea to prove a result on the error control is to observe that the scheme (2.37) (or (2.38) for linear equilibria) and (2.40) come back to utilize the reference scheme $\mathbf{E}$ at the finest level and then perform a truncation, see [Cohen et al., 2003]. Thus we prove, replicating the path of [Cohen et al., 2003], the following statement, which gives a control on the error introduced by the multiresolution lattice Boltzmann adaptive scheme.

Proposition 2.6.1: Additional error estimate

Under Assumptions 2.4.1 and Assumptions 2.6.1, with $\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)$ being the reconstructed solution obtained using the scheme (2.37) and (2.40), the additional error satisfies the following upper bounds:

$$\|\overline{\mathbf{f}}^{\mathrm{ref}}(t) - \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)\| \leq C_{\mathrm{MR}}\epsilon \times \begin{cases} n+1, & \text{if} \quad \tilde{C}_{\mathbf{E}} = 0, \\ 1 + (e^{\tilde{C}_{\mathbf{E}} n} - 1)/\tilde{C}_{\mathbf{E}}, & \text{if} \quad \tilde{C}_{\mathbf{E}} > 0, \end{cases}$$

for $t \in \Delta t\mathbb{N}$.

Therefore, irrespective of the continuity constant $\tilde{C}_{\mathbf{E}}$ of the reference scheme, the additional error is bounded linearly with ϵ. According to the value of the constant $\tilde{C}_{\mathbf{E}}$, we can prove that it accumulates either at most linearly in time or exponentially. It is in general difficult to link the relaxation parameters with the constant and, according to our experience, experiments frequently show a linear behavior even when we expect an exponential one, thus the bound is not sharp.

Proof of Proposition 2.6.1. We start by observing that as stated in the proof of [Cohen et al., 2003, Proposition 4.2] or in [Duarte, 2011, Equation (3.117)] we reconstruct at the finest level both for the collision (2.37) and the stream phase (2.40)

$$\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t + \Delta t) = A_{\Lambda(t+\Delta t)}\mathbf{E}\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t), \tag{2.46}$$

where $\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)$ is reconstructed from the data already adapted on $\Lambda(t + \Delta t)$. Hence by triangle inequality

$$\begin{aligned} \|\overline{\mathbf{f}}^{\mathrm{ref}}(t) - \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)\| &\leq \|\mathbf{E}\overline{\mathbf{f}}^{\mathrm{ref}}(t - \Delta t) - \mathbf{E}\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t - \Delta t)\| + \|\mathbf{E}\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t - \Delta t) - \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t)\| \\ &\leq (1 + \tilde{C}_{\mathbf{E}})\|\overline{\mathbf{f}}^{\mathrm{ref}}(t - \Delta t) - \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t - \Delta t)\| + \|\mathbf{E}\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t - \Delta t) - A_{\Lambda(t)}\mathbf{E}\hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t - \Delta t)\| \\ &\leq (1 + \tilde{C}_{\mathbf{E}})\|\overline{\mathbf{f}}^{\mathrm{ref}}(t - \Delta t) - \hat{\hat{\mathbf{f}}}_{\overline{\ell}}(t - \Delta t)\| + C_{\mathrm{MR}}\epsilon, \end{aligned}$$

employing in this order Assumptions 2.6.1, (2.46), and Assumptions 2.4.1. We have to distinguish two cases and apply the inequality recursively

- $\tilde{C}_{\mathbf{E}} = 0$, thus $\|\bar{\mathbf{f}}^{\mathrm{ref}}(t) - \hat{\hat{\mathbf{f}}}_{\bar{\ell}}(t)\| \le \|\bar{\mathbf{f}}^{\mathrm{ref}}(t-\Delta t) - \hat{\hat{\mathbf{f}}}_{\bar{\ell}}(t-\Delta t)\| + C_{\mathrm{MR}}\epsilon \le \cdots \le C_{\mathrm{MR}}(n+1)\epsilon$. Observe that the term $n+1$ comes from the fact that $\|\bar{\mathbf{f}}^{\mathrm{ref}}(0) - \hat{\hat{\mathbf{f}}}_{\bar{\ell}}(0)\| \neq 0$, but we only have $\|\bar{\mathbf{f}}^{\mathrm{ref}}(0) - \hat{\hat{\mathbf{f}}}_{\bar{\ell}}(0)\| \le C_{\mathrm{MR}}\epsilon$ by virtue of Proposition 2.3.3.

- $\tilde{C}_{\mathbf{E}} > 0$. We obtain, using $(1+\tilde{C}_{\mathbf{E}})^n \le e^{\tilde{C}_{\mathbf{E}} n}$, since $\tilde{C}_{\mathbf{E}} > 0$, that

$$\|\bar{\mathbf{f}}^{\mathrm{ref}}(t) - \hat{\hat{\mathbf{f}}}_{\bar{\ell}}(t)\| \le (1+\tilde{C}_{\mathbf{E}})\|\bar{\mathbf{f}}^{\mathrm{ref}}(t-\Delta t) - \hat{\hat{\mathbf{f}}}_{\bar{\ell}}(t-\Delta t)\| + C_{\mathrm{MR}}\epsilon \le \cdots \le C_{\mathrm{MR}}\epsilon \sum_{i=0}^{n-1}(1+\tilde{C}_{\mathbf{E}})^i + C_{\mathrm{MR}}\epsilon$$
$$\le C_{\mathrm{MR}}\left(1 + \frac{(1+\tilde{C}_{\mathbf{E}})^n - 1}{\tilde{C}_{\mathbf{E}}}\right)\epsilon \le C_{\mathrm{MR}}\left(1 + \frac{e^{\tilde{C}_{\mathbf{E}} n} - 1}{\tilde{C}_{\mathbf{E}}}\right)\epsilon.$$

□

2.7 Boundary conditions

We enforce boundary conditions—which are at the same time physical and numerical—by replacing lacking information in the stream phase, which is the one which might look for information outside the domain. We do not enter in a full discussion on the possible boundary conditions for lattice Boltzmann scheme, see [Krüger et al., 2017, Chapter 5]. Consider a cell $C_{\ell,\boldsymbol{k}}$ touching the boundary of the domain $\partial\Omega$, so that $|C_{\ell,\boldsymbol{k}} \cap \partial\Omega|_{d-1} > 0$ and an index $j \in [\![1, q]\!]$ such that the information comes from outside the domain, that is $\boldsymbol{c}_j \cdot \boldsymbol{n} < 0$ where $\boldsymbol{n}$ is the normal vector to $\partial\Omega$. For illustrative purpose we consider three types of boundary conditions which—for the reference scheme—correspond to

$$\bar{\mathrm{f}}^{j}(t+\Delta t) = \begin{cases} \bar{\mathrm{f}}^{j,\star}(t), & \text{(oth order extrapolation)}, \\ \bar{\mathrm{f}}^{\tilde{j},\star}(t), & \text{(bounce-back)}, \\ -\bar{\mathrm{f}}^{\tilde{j},\star}(t), & \text{(anti-bounce-back)}, \end{cases} \tag{2.47}$$

where $\tilde{j}$ is such that $\boldsymbol{c}_{\tilde{j}} = -\boldsymbol{c}_j$ is the opposite discrete velocity. One can handle non-homogeneous conditions or more intricate ones in the same way. Due to lacking pieces of information that we have isolated in the last term of the following expression and that we note without the reconstruction sign, (2.40) is not well defined:

$$\bar{\mathrm{f}}^{j}_{\ell,\boldsymbol{k}}(t+\Delta t) = \bar{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}}(t) + \frac{1}{2^{d\Delta\ell}}\Bigg(\sum_{\bar{\boldsymbol{k}} \in \mathcal{E}^{j}_{\ell,\boldsymbol{k}} \cap [\![0, N_{\bar{\ell}}]\!]^d} \hat{\hat{\mathrm{f}}}^{j,\star}_{\bar{\ell},\bar{\boldsymbol{k}}}(t) - \sum_{\bar{\boldsymbol{k}} \in \mathcal{A}^{j}_{\ell,\boldsymbol{k}}} \hat{\hat{\mathrm{f}}}^{j,\star}_{\bar{\ell},\bar{\boldsymbol{k}}}(t)\Bigg) + \underbrace{\frac{1}{2^{d\Delta\ell}} \sum_{\bar{\boldsymbol{k}} \in \mathcal{E}^{j}_{\ell,\boldsymbol{k}} \setminus [\![0, N_{\bar{\ell}}]\!]^d} \bar{\mathrm{f}}^{j,\star}_{\bar{\ell},\bar{\boldsymbol{k}}}(t)}_{\text{(not well defined)}}.$$

Observe that having for reasonably small velocity stencils (all the stencils we consider in the work, namely those implying at most two neighbors in each direction), $\mathcal{A}^{j}_{\ell,\boldsymbol{k}} \subset \mathcal{B}_{\ell,\boldsymbol{k}} \subset [\![0, N_{\bar{\ell}}]\!]^d$, all the quantities appearing in the outgoing pseudo-fluxes are known since they come from inside the domain. Using (2.47) and presenting the "bounce-back" condition for the sake of presentation, we obtain

$$\bar{\mathrm{f}}^{j}_{\ell,\boldsymbol{k}}(t+\Delta t) = \bar{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}}(t) + \frac{1}{2^{d\Delta\ell}}\Bigg(\sum_{\bar{\boldsymbol{k}} \in \mathcal{E}^{j}_{\ell,\boldsymbol{k}} \cap [\![0, N_{\bar{\ell}}]\!]^d} \hat{\hat{\mathrm{f}}}^{j,\star}_{\bar{\ell},\bar{\boldsymbol{k}}}(t) - \sum_{\bar{\boldsymbol{k}} \in \mathcal{A}^{j}_{\ell,\boldsymbol{k}}} \hat{\hat{\mathrm{f}}}^{j,\star}_{\bar{\ell},\bar{\boldsymbol{k}}}(t)\Bigg) + \underbrace{\frac{1}{2^{d\Delta\ell}} \sum_{\bar{\boldsymbol{k}} \in (\mathcal{E}^{j}_{\ell,\boldsymbol{k}} \setminus [\![0, N_{\bar{\ell}}]\!]^d) + \boldsymbol{c}_j} \bar{\mathrm{f}}^{\tilde{j},\star}_{\bar{\ell},\bar{\boldsymbol{k}}}(t)}_{\text{(well defined)}},$$

where now all sums imply quantities inside the domain Ω. The distribution in the last sum can be evaluated in two ways. The first one uses the usual reconstruction operator. The second one is using a direct evaluation, which is a cheaper and easier alternative. We implement the latter method, which consists in taking the value directly available on the ancestor cell present in $S(\Lambda(t+\Delta t))$.

2.8 Numerical tests

In Section 2.8, we test our method in order to showcase that it indeed meets the requirements that we have previously fixed in the Introduction of Chapter 2. For the one dimensional setting $d = 1$, we focus on

- Retrieving the theoretical error estimates.
- Reduce the memory occupation.
- Test for many schemes and parameters, in order to show the generality of the approach.
- Study the influence of the collision strategy on the quality of the solution.

For the two and three dimensional settings $d = 2, 3$, more emphasis is placed onto

- Retrieving the theoretical error estimates, in particular for a non-linear hyperbolic problem.
- Use the approach with the most classical lattice Boltzmann scheme, namely the D_2Q_9 scheme with three conserved moments under acoustic scaling, and monitor macroscopic quantities (*e.g.* drag coefficient, *etc.*)
- The possibility of using the strategy in 3D to yield an extremely reduced memory footprint.

All the computations are done under acoustic scaling, thus when $\lambda > 0$ remains fixed as $\Delta x \to 0$ (*i.e.* $\overline{\ell} \to +\infty$). Unless otherwise stated, the test in Section 2.8 are carried using the "leaves collision" by (2.38). An exception to this rule—where the "reconstructed collision" (2.37) is used to explain a pathological case—is explicitly highlighted. The same will happen to compare "reconstructed collision" (2.37) , "leaves collision" (2.38) and "predict-and-integrate collision" (2.39). Moreover, we consider the number of neighbors taken by the prediction operator $\boldsymbol{P}_\triangle$ to be $\gamma = 1$, except for some selected tests where $\gamma = 2$ is also considered with explicit mention. Larger prediction stencils entail larger costs of the multiresolution analysis and thus a larger overhead. However, since the decay of the details is linked to γ *via* (2.26), for very smooth solutions, it can be beneficial to increase γ in order to achieve very high compression factors. Quite the opposite, for solutions with shocks, the details do not decay with the level ℓ whatever γ is. It is therefore not advisable to use large γ in this situation, because this does not yield important gains in the mesh compression compared to the larger overhead.

2.8.1 1D tests

2.8.1.1 Metrics and setting

In Section 2.8.1, we focus on the following three aspects.

- The fulfillment of the theoretical estimate by Proposition 2.6.1. The errors are measured on the conserved moments, which are the variables of interest for which exact solutions for the continuous problem can be defined. The norm of choice is the ℓ^1-norm. We look at

$$E^{i,n} := \|\overline{\boldsymbol{m}}^{\mathrm{ex},i}(n\Delta t) - \overline{\mathbf{m}}^{\mathrm{ref},i}(n\Delta t)\|_{\ell^1}, \qquad e^{i,n} := \|\overline{\boldsymbol{m}}^{\mathrm{ex},i}(n\Delta t) - \hat{\overline{\mathbf{m}}}^{i}_{\overline{\ell}}(n\Delta t)\|_{\ell^1}, \tag{2.48}$$

$$\delta^{i,n} := \|\overline{\mathbf{m}}^{\mathrm{ref},i}(n\Delta t) - \hat{\overline{\mathbf{m}}}^{i}_{\overline{\ell}}(n\Delta t)\|_{\ell^1}, \tag{2.49}$$

  for $i \in [\![1, N]\!]$ spanning the conserved moments and $n \in [\![0, n_T]\!]$ where $n_T \in \mathbb{N}$ is such that $n_T \Delta t = T$ with $T > 0$ the final time of the simulation. These metrics are respectively: $E^{i,n}$, the error of the reference method against the exact solution, called "reference discretization error"; $e^{i,n}$, the error of the adaptive method against the exact solution, called "adaptive discretization error"; $\delta^{i,n}$, the difference between the adaptive solution and the reference solution, called "perturbation error".

  As seen in Proposition 2.6.1, the perturbation error $\delta^{i,n} \sim \epsilon$. By the triangle inequality, we have that

$$e^{i,n} \leq E^{i,n} + \delta^{i,n}.$$

  An important aspect once utilizing multiresolution, linked with the choice of ϵ, is not to perturb the reference discretization error due to the perturbation error, that is having $\delta^{i,n} \ll E^{i,n}$. This is independent of the fact

that the reference scheme is convergent, namely $E^{i,n} \to 0$ as $\overline{\ell} \to +\infty$. Clearly, for convergent schemes, if the user increases $\overline{\ell}$, the threshold parameter ϵ has to be decreased accordingly in order to avoid interference with the convergence of the scheme, thus to have $\delta^{i,n} \ll E^{i,n}$ entailing $e^{i,n} \simeq E^{i,n}$.

- The gain in terms of memory occupation and—at the very end—computational time induced by the use of multiresolution. In Section 2.8.1, we use the compression rate

$$\mathrm{CR}^n := 100 \times \left(1 - \frac{\#(S(\Lambda(n\Delta t))}{N_{\overline{\ell}}}\right),$$

also as a measure of computational efficiency, knowing that the real one is strongly dependent on the implementation and data structure.

We also use the time-averaged compression rate given by

$$\mathrm{ACR}^{n_T} := 100 \times \left(1 - \frac{\frac{1}{n_T}\sum_{n=1}^{n=n_T} \#(S(\Lambda(n\Delta t))}{N_{\overline{\ell}}}\right).$$

In what we present next, the metric ACR^{n_T} is generally bounded from below by the compression factor at the final time CR^{n_T} for the following reason. We mostly start from solutions with shocks, where very high compression rates are achieved. Eventually, we obtain traveling shocks, contact discontinuities, and rarefactions fans, thus having to put more and more cells and worsening the compression rate. Thus, the compression rate at the final time clearly bounds the average compression rate from below.

- Compare different collision strategies, namely (2.37), (2.38) and (2.39), in terms of quality of the obtained solution.

In Section 2.8.1, we consider $\epsilon = 1\mathrm{e}-4$, $\underline{\ell} = 2$ and $\overline{\ell} = 9$, except when these parameters are varied or otherwise said. This value for ϵ guarantees, for any considered test with reference scheme at level $\overline{\ell} = 9$, to achieve $\delta^{i,n} \ll E^{i,n}$. Notice that $\overline{\ell}$ determines which reference scheme we are relying on and building our adaptation strategy. Differently, $\underline{\ell}$ can be chosen freely by the user and determines the number of potential levels in the adaptive mesh.

2.8.1.2 D_1Q_2 for one conservation law

We start with the simplest lattice Boltzmann scheme, with a first problem being linear and a second one that can develop blowups and shocks.

2.8.1.2.1 The problem and the scheme We consider the approximation of the weak entropic solution [Serre, 1999] of the initial-value problem for the scalar conservation law

$$\begin{cases} \partial_t u + \partial_x(\varphi(u)) = 0, & t \in [0,T], \quad x \in \mathbb{R}, \\ u(0,x) = u^\circ(x), & x \in \mathbb{R}, \end{cases} \tag{2.50}$$

with a flux $\varphi \in C^\infty(\mathbb{R})$ and an initial datum $u^\circ \in L^\infty(\mathbb{R})$. This problem is the advection equation with constant velocity V for $\varphi(u) = Vu$ and the inviscid Burgers equation for $\varphi(u) = u^2/2$.

We use the D_1Q_2 with $N = 1$ introduced in Section 1.5.1, under acoustic scaling. With the theory of the equivalent equations [Dubois, 2008], [Graille, 2014] has shown that this scheme is first order consistent in Δx with (2.50) upon selecting $m_2^{\mathrm{eq}}(\mathsf{m}_1) = \varphi(\mathsf{m}_1)$, with m_1 which will be an approximation of u, the solution of (2.50).

Remark 2.8.1 (Continuity constant of the scheme)**.** *Observe that in the case of advection equation, that is when $\varphi(u) = Vu$ for $0 < V \le \lambda$ (CFL condition), we can explicitly determine the continuity constant of the reference scheme $\mathbf{E}$ for the L^1 norm, assuming to deal with periodic boundary conditions. Indeed, the continuity constant $C_{\mathbf{E}} \ge 1$ is the induced L^1 norm of $\mathbf{E}$, which thus reads*

$$C_{\mathbf{E}} = \max\left\{\left|1 - \frac{s_2}{2} - \frac{s_2 V}{2\lambda}\right| + \left|\frac{s_2}{2} + \frac{s_2 V}{2\lambda}\right|, \left|1 - \frac{s_2}{2} + \frac{s_2 V}{2\lambda}\right| + \left|\frac{s_2}{2} - \frac{s_2 V}{2\lambda}\right|\right\}$$

Table 2.3: Test cases for one scalar conservation law with choice of flux φ, initial datum u°, expected regularity $W^{\nu,\infty}$ of the solution, influencing the choice of regularity parameter $\overline{\mu} = \min(\nu, 2\gamma+1)$ for the refinement criterion (*cf.* Section 2.4.2) and final time T of the simulation.

Flux φ	Initial datum u°	Type of solution	ν	T	Test nb.
$\varphi(u) = \frac{3}{4}u$	$u^\circ(x) = e^{-20x^2}$	Strong C^∞	∞	0.4	(I)
	$u^\circ(x) = \mathbb{1}_{\lvert x\rvert\le 1/2}(x)$	Weak L^∞	0	0.4	(II)
$\varphi(u) = \frac{u^2}{2}$	$u^\circ(x) = (1+\tanh(100x))/2$	Strong C^∞	∞	0.4	(III)
	$u^\circ(x) = \mathbb{1}_{\lvert x\rvert\le 1/2}(x)$	Weak L^∞	0	0.7	(IV)
	$u^\circ(x) = (1+x)\mathbb{1}_{x<0}(x) + (1-x)\mathbb{1}_{x\ge 0}(x)$	Weak L^∞	0	1.3	(V)

$$= \max\left\{\left|1 - \frac{s_2}{2} - \frac{s_2 V}{2\lambda}\right| + \frac{s_2}{2} + \frac{s_2 V}{2\lambda}, 1\right\} = \begin{cases} 1, & \text{if} \quad s_2 \le 2/(1+V/\lambda), \\ s_2(1+V/\lambda) - 1, & \text{if} \quad s_2 > 2/(1+V/\lambda). \end{cases}$$

We shall consider five test cases as given in Table 2.3 and a computational domain $\Omega = [-3,3]$. We endow the scheme with oth order extrapolation boundary conditions. We fix $\lambda = 1$ and we consider different relaxation parameters s_2.

2.8.1.2.2 Results We first vary the threshold parameter ϵ and monitor the the evolution of the perturbation error δ^{1,n_T} as well as the compression factors CR^{n_T}, ACR^{n_T} at final time. Moreover, we fix $\epsilon = 1\mathrm{e}{-4}$ as previously claimed and we monitor the time evolution of $\delta^{1,n}$, $E^{1,n}/\delta^{1,n}$, $E^{1,n}/e^{1,n}$ and CR^n, respectively the perturbation error, the ratio of reference and perturbation error, the ratio of reference and adaptive error, the compression rate.

(I) The results are available in Figure 2.10. We observe that with this choice of ϵ we successfully keep the perturbation error $\delta^{1,n}$ about 10 to 100 times smaller than the reference discretization error $E^{1,n}$ at the chosen level $\overline{\ell}$, with important compression rates around 95% for the chosen ϵ. We note the fairly correct linear behavior in terms of ϵ. The compression factor CR^{n_T} and the average compression factor ACR^{n_T} coincide because the solution retains the same smoothness in time and is simply transported (plus the numerical diffusion). The perturbation error increases linearly in time even when we can only prove an exponential bound, *cf.* Proposition 2.6.1 and Remark 2.8.1, when $s_2 > 8/7$. Concerning the ratio $E^{1,n}/\delta^{1,n}$, one should remark that we have a boundary layer close to the initial time, tending to small values because we are dealing with regular solutions. Indeed, many unrefined areas where the approximation made during the stream phase generates, from the very beginning, an adaptive scheme which is quite different from the reference scheme. Therefore, for small n, we have either $\delta^{1,n} \simeq E^{1,n}$ or maybe also $\delta^{1,n} \gg E^{1,n}$. Still, even in this case, as long as the time grows, we are capable of largely outperform against the reference scheme, yielding $\delta^{1,n} \ll E^{1,n}$ for large n. For the choice of $\overline{\ell}$ and threshold parameter ϵ we made, the ratio $E^{1,n}/e^{1,n}$ remains very close to one for any considered time.

(II) The results are given in Figure 2.11. The perturbation error of the adaptive method is about three orders of magnitude smaller than the reference discretization error. Due to the presence of large *plateaux*, the compression factor is really interesting for a large range of ϵ, being always over 90%. The trend of δ^{1,n_T} as a function of ϵ agrees with the theory. We see that ACR^{n_T} is larger than CR^{n_T} arguably because of the numerical diffusion which accumulates in time and smears the shock. Again, the perturbation error $\delta^{1,n}$ increases linearly in time regardless of the choice of relaxation parameter s_2. Looking at the ratio $E^{1,n}/\delta^{1,n}$, we observe a boundary layer close to the initial time with large values tending to $+\infty$. This can be understood by the fact that at the beginning, since working with Riemann problems, we have added enough security cells around the shock with the refinement $\mathcal{H}_\epsilon$ and the grading $\mathcal{G}$—*cf.* (2.34)—in order to make the adaptive scheme "degenerate" to the reference scheme, thus $\delta^{1,n} \ll E^{1,n}$ for small n. Finally, for the choice of $\overline{\ell}$ and threshold parameter ϵ at hand, the ratio $E^{1,n}/e^{1,n}$ remains very close to one for any considered time.

(III) The results are in Figure 2.12. Again, we observe that this choice of ϵ guarantees perturbation errors which are between 5 and 50 times smaller than the discretization error of the reference method, still preserving excellent compression rates. We again have $\mathrm{ACR}^{n_T} > \mathrm{CR}^{n_T}$ because of the formation of a rarefaction fan

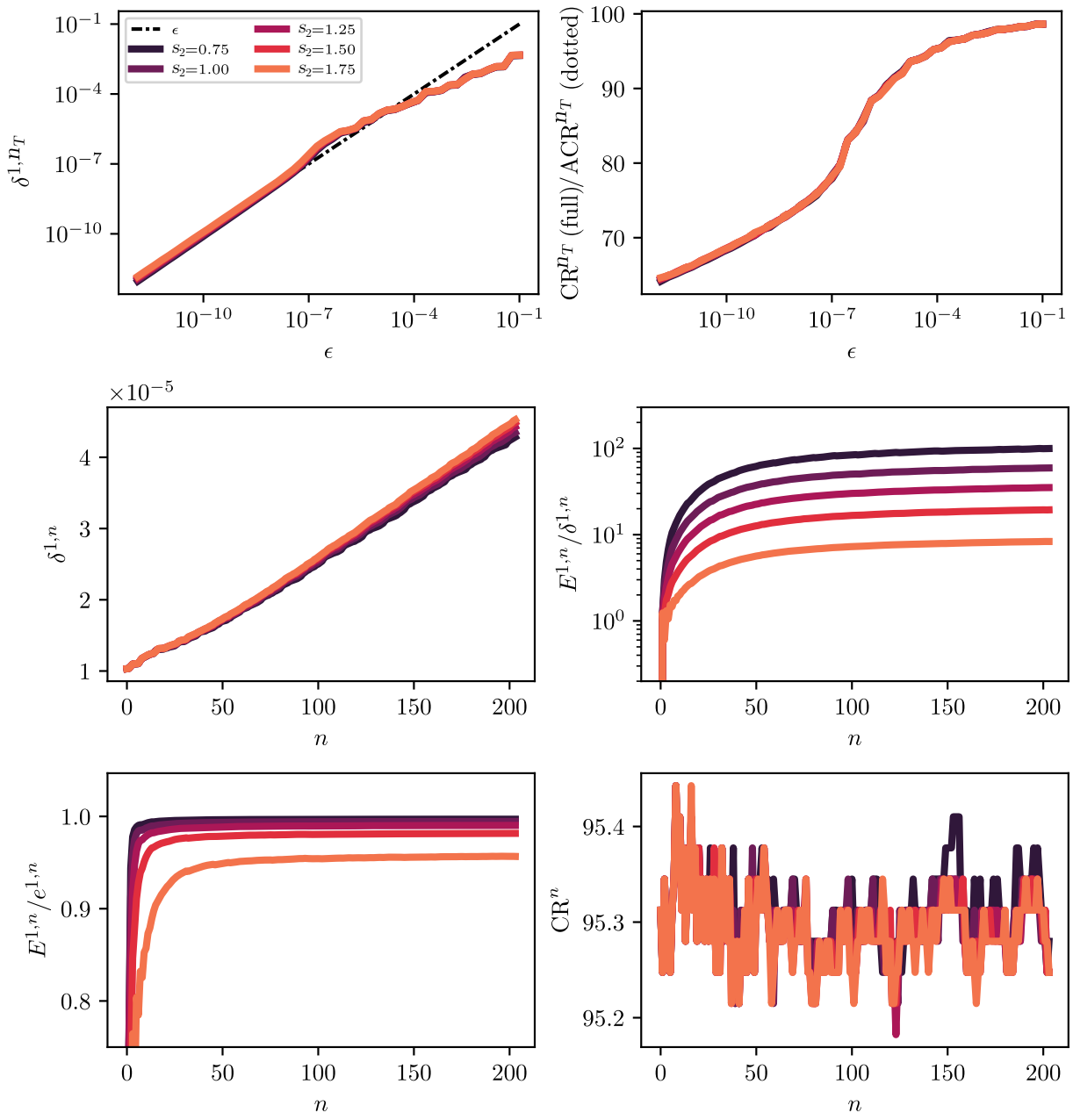


Figure 2.10: Test (I). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

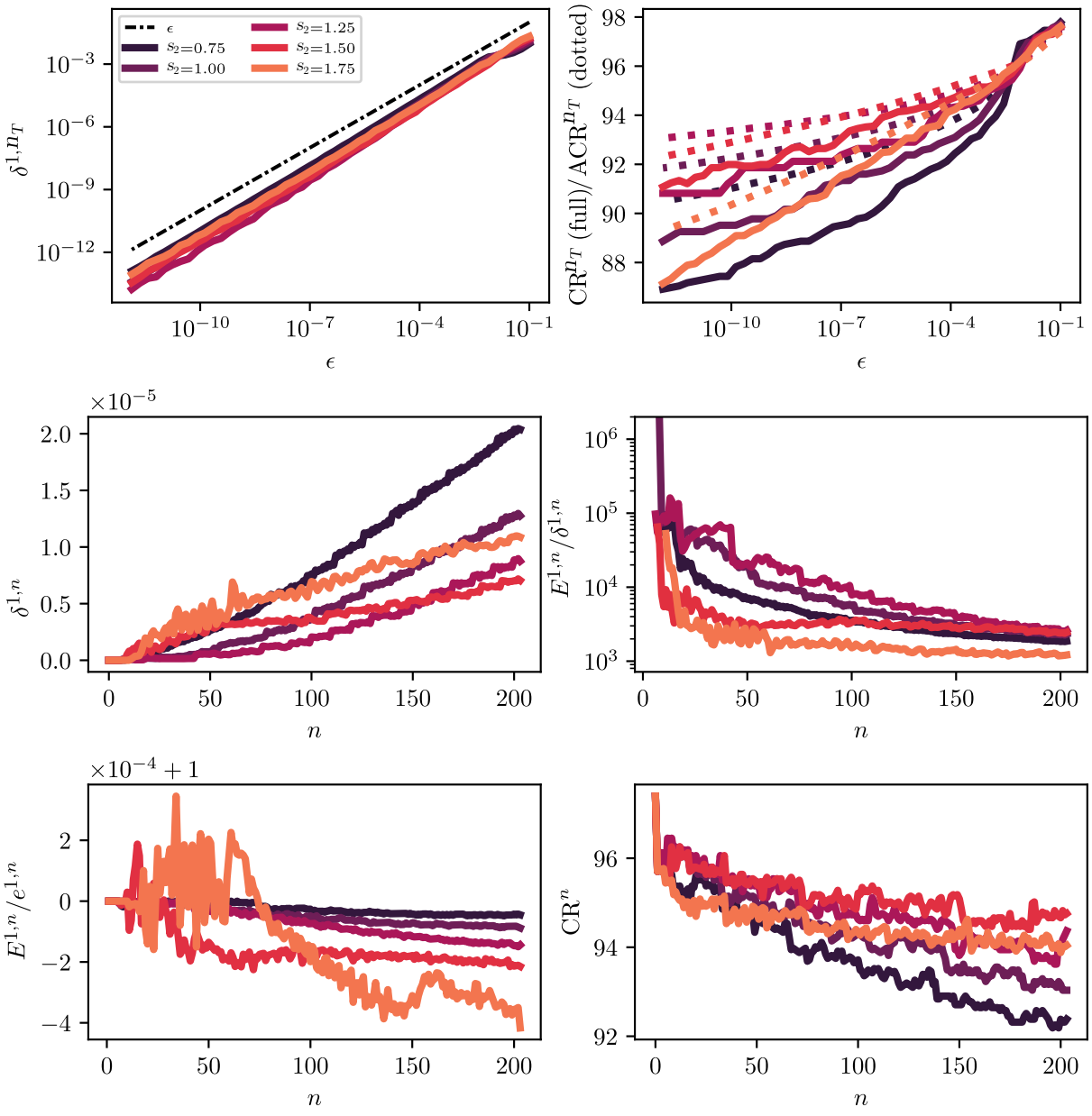


Figure 2.11: Test (II). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

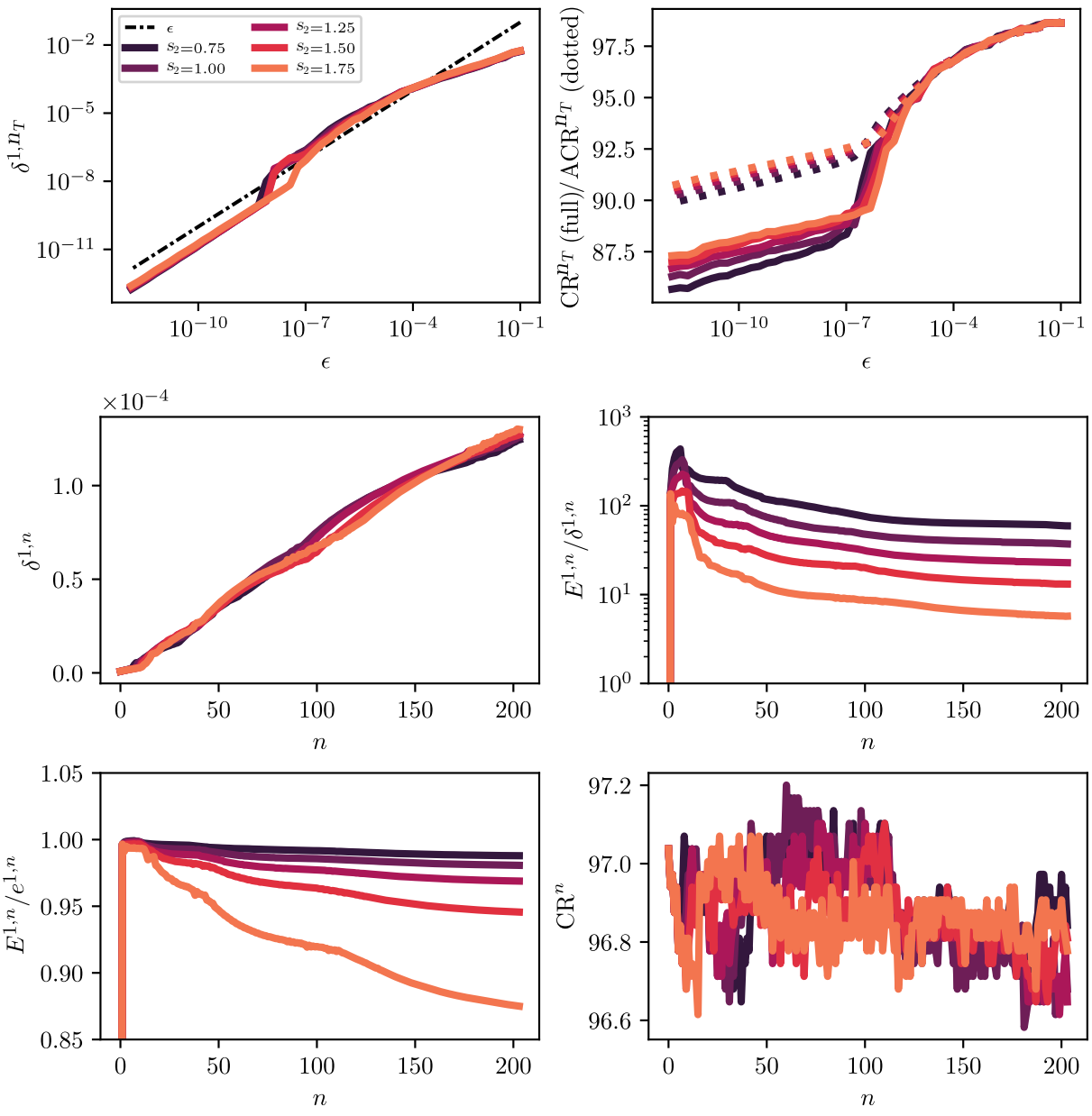


Figure 2.12: Test (III). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

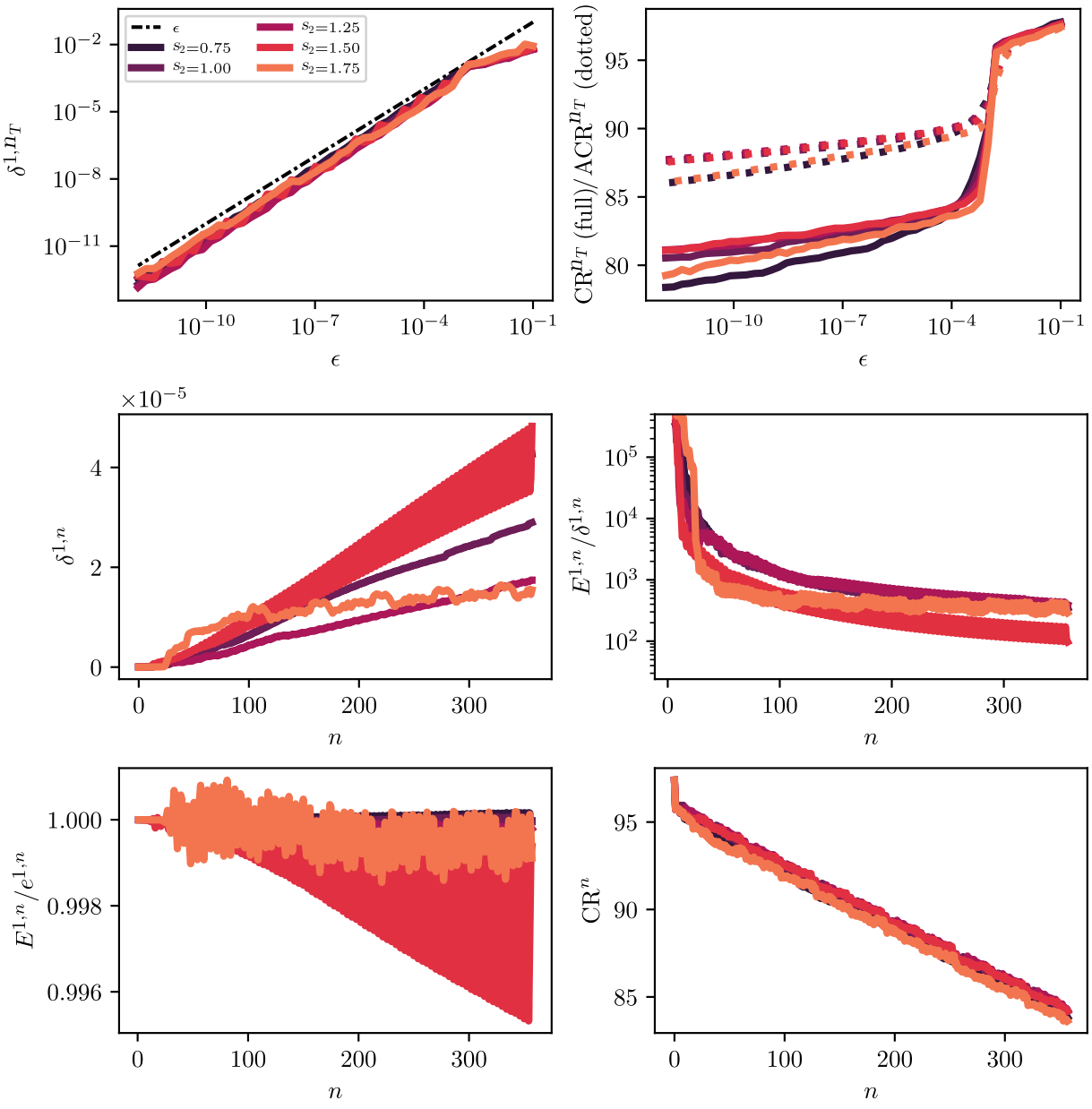


Figure 2.13: Test (IV). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

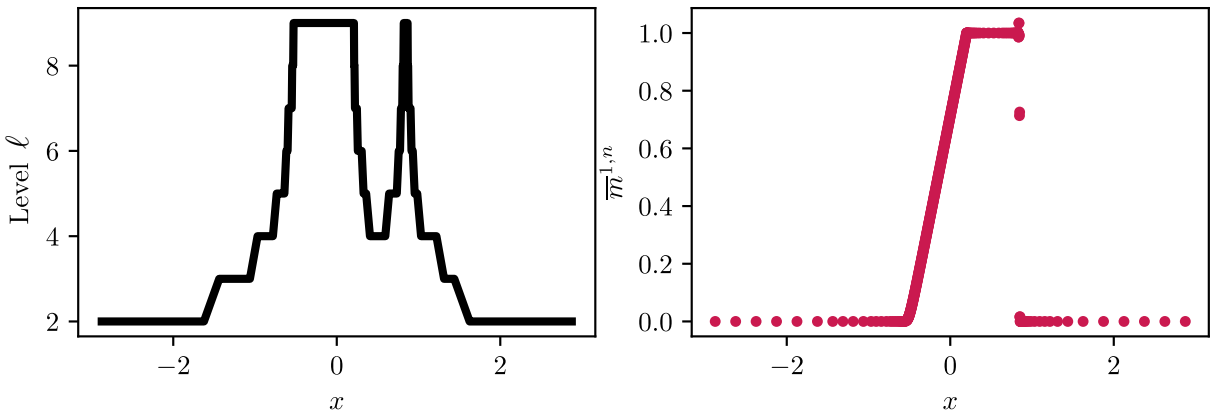


Figure 2.14: Example of solution from the adaptive scheme for the test (IV), considering $n = 358$, $s_2 = 1.5$ and $\epsilon = 1\mathrm{e}-4$ On the left, levels of the computational mesh. On the right, solution on the leaves of the tree.

as the simulation goes on. The behavior as ϵ tends to zero is respected and the expected linear temporal trend is obtained. The attentive reader could have observed the following fact: the compression rates tend to stagnate as $\epsilon \to 0$. The reason for that is the following (and applies to any context in which the situation shall happen, also in what follows): consider the typical solution of most of the problems we consider, where only shocks (and contact discontinuities) and rarefaction fans are present. Elsewhere, the solution is essentially flat. Start from a very large threshold ϵ: multiresolution does not put cells at the finest level of resolution $\overline{\ell}$ because the threshold is really large. Then decrease ϵ little by little: the finest resolution is reached on the shock and in the less smooth zones of the rarefaction fans. By continuing decreasing ϵ, the fans are also refined (especially if here the solution is highly nonlinear). Nevertheless, at some time, the finest level $\overline{\ell}$ is reached everywhere where the solution is non-flat (shocks and fans) and eventually (for smaller ϵ) there is not so much room for improving the quality of the reconstruction by refining elsewhere, because here the solution is totally flat (and indeed the details are perfectly equal to zero). This is why the compression rate (almost) stagnates. Multiresolution can still diminish the error as expected by adding very few cells thus with very little modifications of the compression rates. Of course, one expects $\mathrm{CR}^{n_T}, \mathrm{ACR}^{n_T} \to 0$ as $\epsilon \to 0$, but in this case ϵ should become really small, presumably below the machine-epsilon to observe the convergence after the stagnation. For the ratio $E^{1,n}/\delta^{1,n}$, the same remarks as (I) apply.

(IV) The results are in Figure 2.13 and—for illustrative purpose—the numerical solution of the adaptive scheme is shown in Figure 2.14. The adaptive method largely beats the traditional method by three orders of magnitude when comparing the perturbation error to the error of the reference scheme, with less efficient compression compared to (II) due to the formation of a rarefaction fan which—though straight-shaped—is refined by multiresolution even away from the extremal kinks of the slopes because the $\mathrm{D}_1\mathrm{Q}_2$ exhibits checkerboard patterns in this area. Again, this causes the fact that $\mathrm{ACR}^{n_T} > \mathrm{CR}^{n_T}$. The estimate in ϵ is sharply met and the perturbation error increases linearly in time for every choice of relaxation parameter s_2. Strong oscillations due to the reference scheme are present close to the shock, especially when using a relaxation parameter $s_2 > 1$. For the ratio $E^{1,n}/\delta^{1,n}$, the same remarks as (II) apply. The compression rate CR^n grows linearly in time because the size of the rarefaction fan grows linearly in time as well.

(V) The results are in Figure 2.15. This test provides a pathological and *ad-hoc* example where the reconstructed collision (2.37) is needed instead of the leaves collision (2.38) to correctly retrieve the theoretical estimates on the perturbation error $\delta^{1,n}$ controlled by the threshold ϵ. This is due to the fact that the solution is piece-wise linear for every time—especially at initial time—and we know that the prediction operator with $\gamma = 1$ is exact (*cf.* Proposition 2.3.1) on each linear branch of the solution. Remark that the weak solution blows up at time $T^\star = 1$ and we take $\overline{\mu} = 0$ in order to be sure of correctly capture the jump in the solution after this event. Moreover, the final time is taken to $T = 1.3$ to observe the blowup. Looking at Figure 2.15 three notable facts are to remark and arise from the particular way of fabricating the solution. The first is

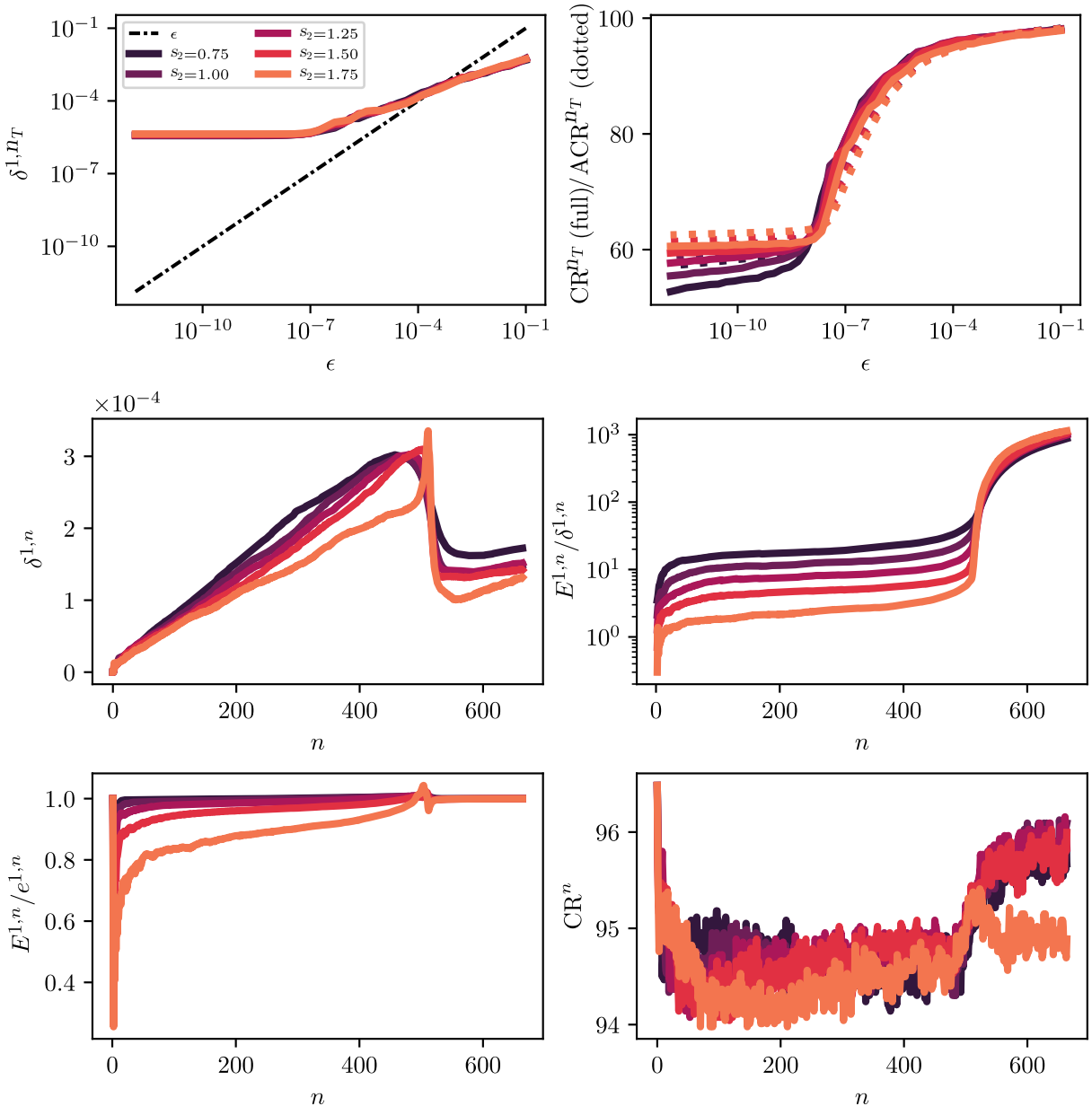


Figure 2.15: Test (V). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

that the temporal trend of the perturbation error $\delta^{1,n}$ changes at the blowup time $T^\star = 1$ (corresponding to $n^\star = 512$). This is coherent with the fact that the solution changes its regularity from $W^{1,\infty}$ to just L^∞ (consider the effect of the details given by Proposition 2.3.2), whereas the threshold ϵ to which details are compared whilst applying $\mathcal{T}_\epsilon$ and $\mathcal{H}_\epsilon$ (*cf.* (2.34)) is kept fixed in time. Second, the ratio $E^{1,n}/\delta^{1,n}$ shows a time boundary layer close to $n = 0$, tending towards small values. This means that at the very beginning of the simulation, the error of the reference scheme is comparable (or smaller) to that of the adaptive scheme, as we already observed for case (I) and (III). This fact shall be explained in a moment and we will not come to the same conclusions as for case (I) and (III) concerning the dominant causes of this phenomenon. Lastly, we observe that after an initial decrease, δ^{1,n_T} stagnates as ϵ decreases as well as the compression factor. This is in contradiction with the theoretical estimates which give $\delta^{1,n_T} \lesssim \epsilon$. However, one should not forget that we have used the "leaves collision" instead of the "reconstructed collision" and this test case has been built on purpose to obtain this.

We now provide a full explanation for these remarks, as well as an additional test. Since the initial solution is piece-wise linear, the multiresolution analysis $\mathcal{T}_\epsilon$ and the grading $\mathcal{G}$ put more and more cells close to the kinks (located at $x = -1, 0, 1$) as ϵ decreases, until reaching a point where the prediction $\boldsymbol{P}_\triangle$ (and thus the reconstruction) is exact and no more cell have to be added. As the reconstruction process pertains to the advection phase, from a certain ϵ and at the beginning of the simulation, the stream is exact: the same as the reference scheme. This is false for the collision, because of the non-linearity of the collision phase (generated by the non-linear flux $\varphi(u) = u^2/2$ pertaining to the Burgers equation). Along the sloped sides of the hat (between $[-1, 0]$ and $[0, 1]$), the collision on the leaves adds, at the very beginning of the simulation, an error which is the same for all the ϵ smaller than a certain threshold—because the initial grid is indeed the same—and which remains for the whole simulation, yielding the saturation. We have observed exactly the same saturation as ϵ decreases, outside the context of lattice Boltzmann schemes, just by compressing the mesh by multiresolution based on the initial datum, performing the evaluation of the function $\varphi(u)$ on the leaves and measuring the error compared to the evaluation of the function $\varphi(u)$ on the full mesh at the finest level $\overline{\ell}$.

To corroborate our observations, we use the reconstructed collision: in this case, we recover the right estimate in ϵ, see Figure 2.17. This happens because the reconstruction at the finest level is exact on the slopes of the hat and thus the collision has been evaluated at the right resolution. Moreover, the behavior of the initial boundary layer on the plot concerning the ratio $E^{1,n}/\delta^{1,n}$ has been reversed, yielding large values $E^{1,n}/\delta^{1,n} \gg 1$ for small n. This is coherent with the other simulations with weak solutions (tests (II) and (IV)), where at the beginning, the perturbation error $\delta^{1,n}$ is largely negligible compared to $E^{1,n}$ but is different for what happened for the regular test (III), where we have checked, switching from the leaves collision Figure 2.12 to the reconstructed collision Figure 2.16 does not change this initial boundary layer. Therefore, we can claim that in the setting of test (V), the dominant phenomenon causing the initial boundary layer is the leaves collision, and not a combination of stream phase and the collision phase (no matter how it is done) as for test (III). Indeed, if we compare Figure 2.15 and Figure 2.17, we notice that the tangent to the curve in the origin is way less steep in the latter case than in the former. On the other hand, for the test (III) in Figure 2.12 and Figure 2.16 the tangent to $\delta^{1,n}$ close to $n = 0$ behaves gently both in the case of leaves collision and reconstructed collision. In the case of test (V), in the case of leaves collision the perturbation error $\delta^{1,n}$ is about one order of magnitude larger than in the case of reconstructed collision. This is not the case for test (III), where we have only a factor two between the errors using the leaves collision and the reconstructed collision.

To conclude, we have devised a particular case where the "reconstructed collision" (2.37) is needed instead of the "leaves collision" (2.38) to recover the theoretical estimates. Of course, this does not prevent us from having very interesting ratios $E^{1,n}/\delta^{1,n}$ far from the initial time for both cases. In the vast majority of the cases, the leaves collision is largely sufficient and does not prevent one from observing the theoretical behavior.

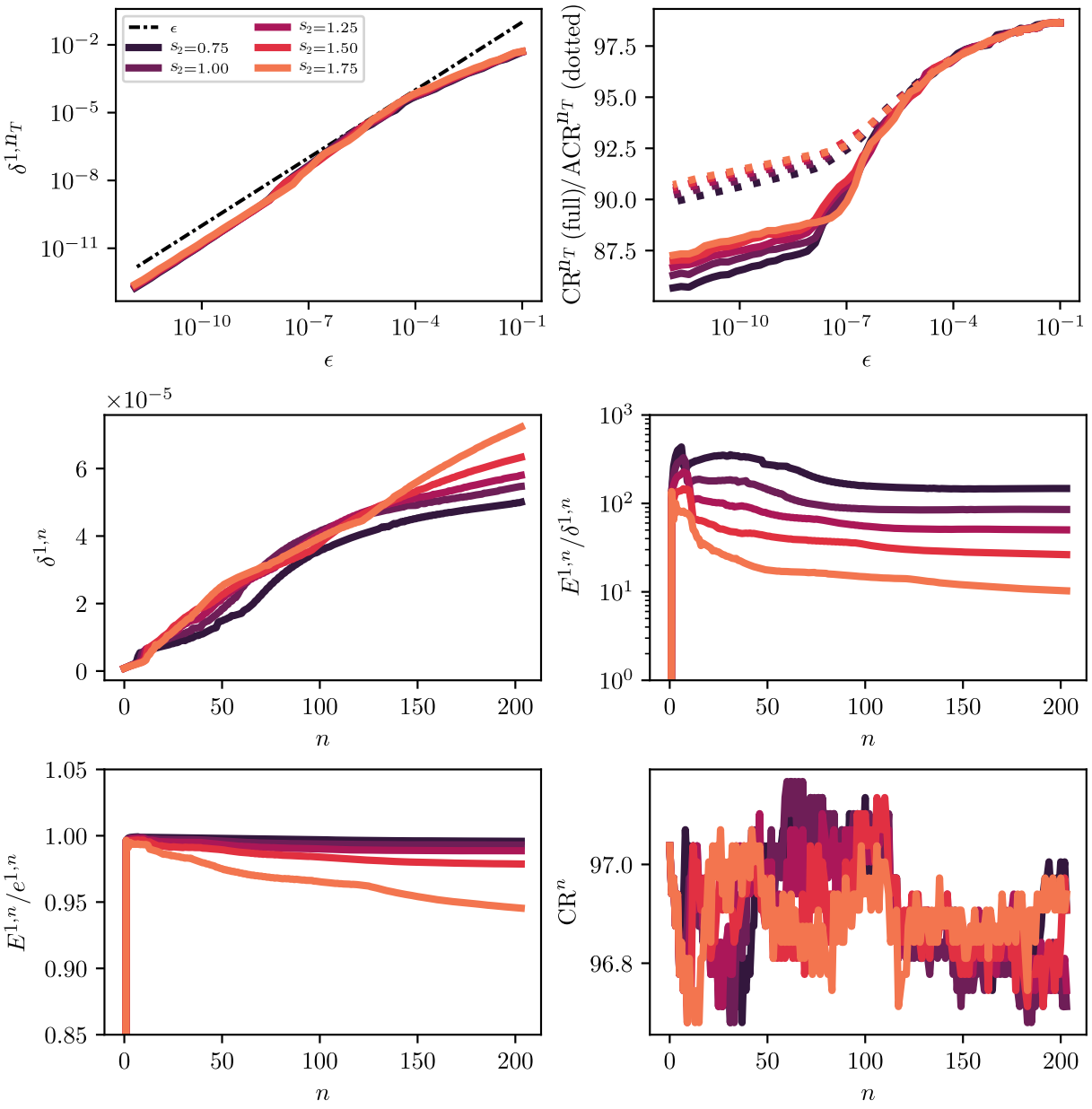


Figure 2.16: Test (III) repeated using the "reconstructed collision" (2.37). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

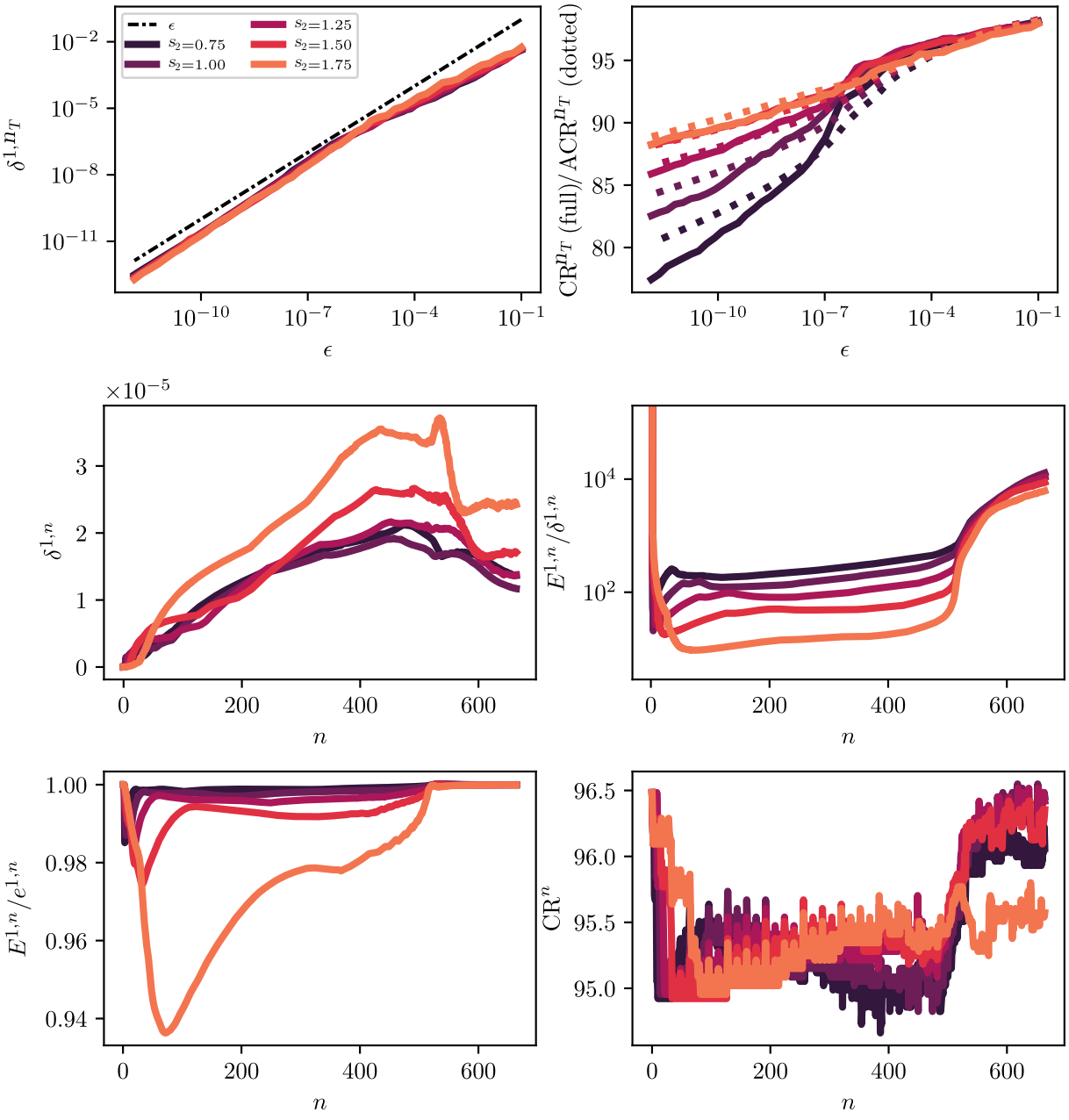


Figure 2.17: Test (V) repeated using the "reconstructed collision" (2.37). First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time.

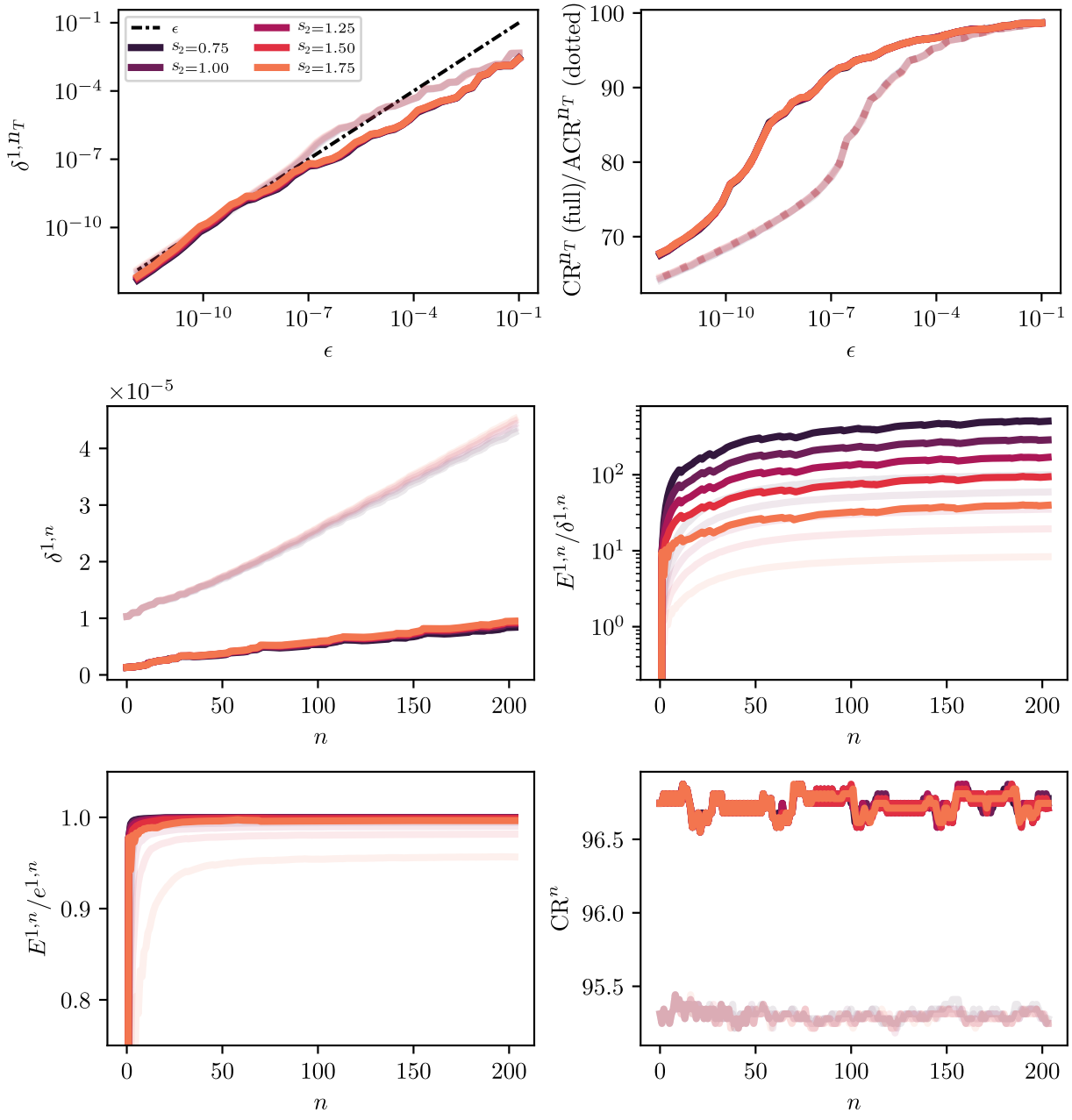


Figure 2.18: Test (I) using $\gamma = 2$. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. The superimposed transparent lines refer to $\gamma = 1$ for comparison.

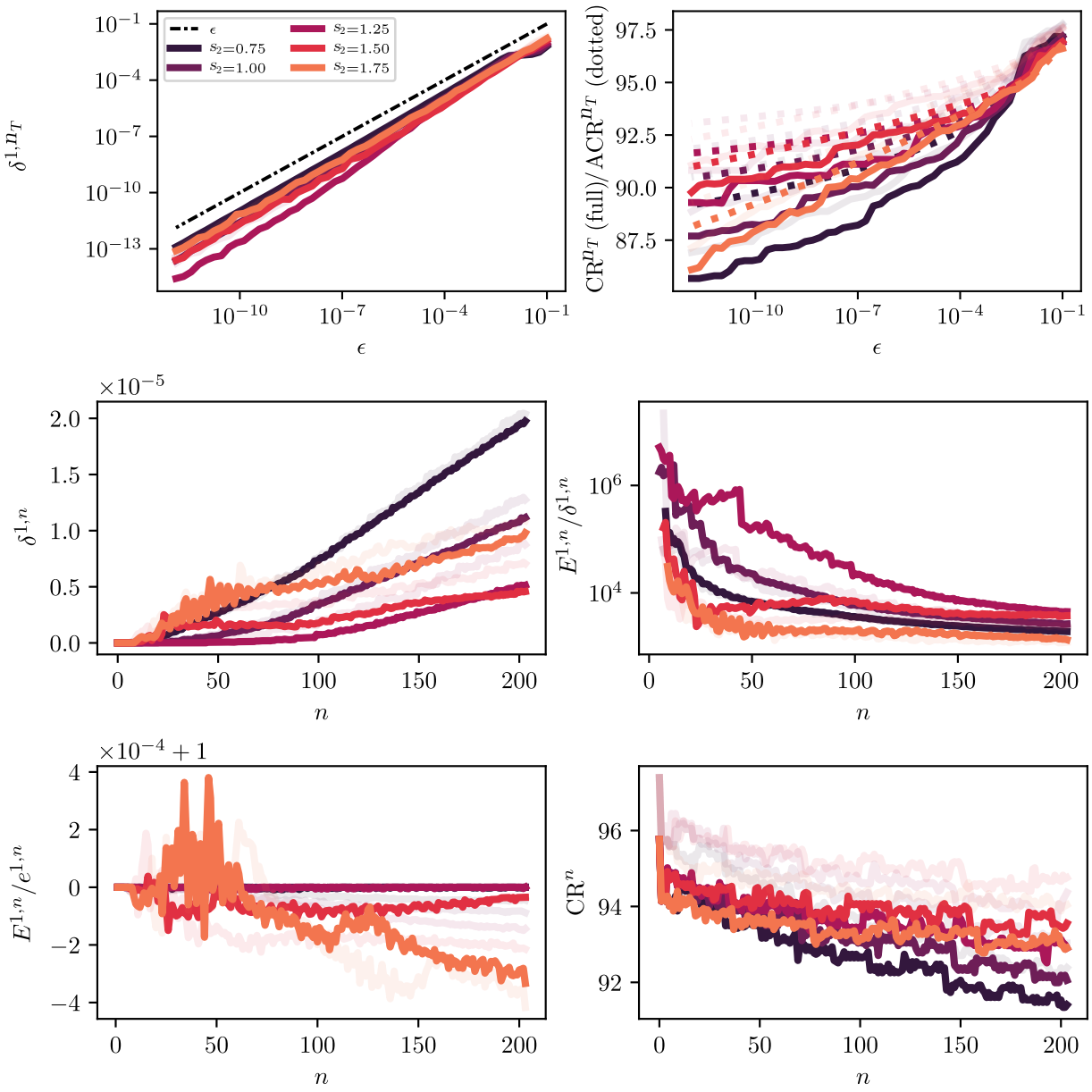


Figure 2.19: Test (II) using $\gamma = 2$. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. The superimposed transparent lines refer to $\gamma = 1$ for comparison.

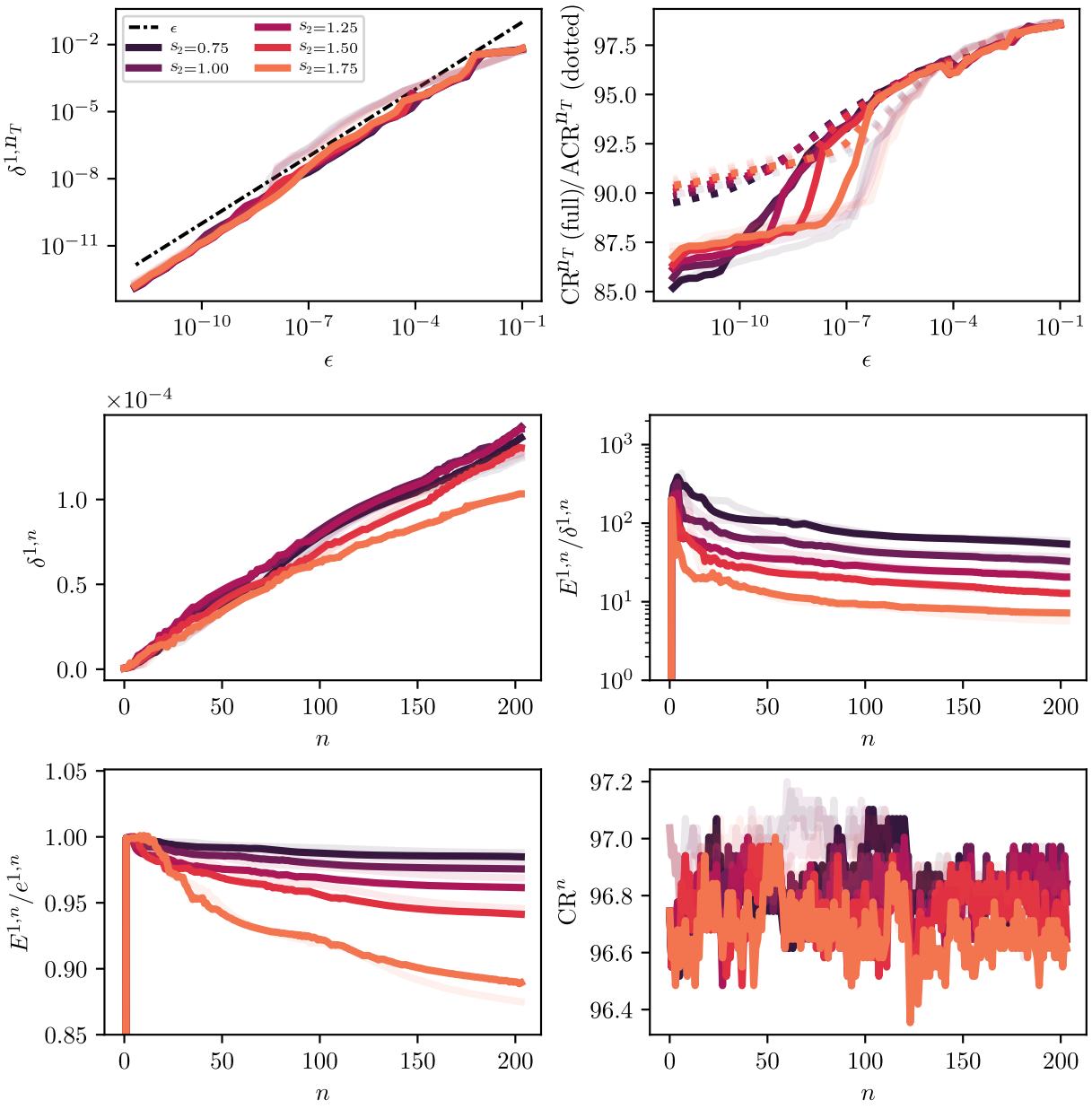


Figure 2.20: Test (III) using $\gamma = 2$. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. The superimposed transparent lines refer to $\gamma = 1$ for comparison.

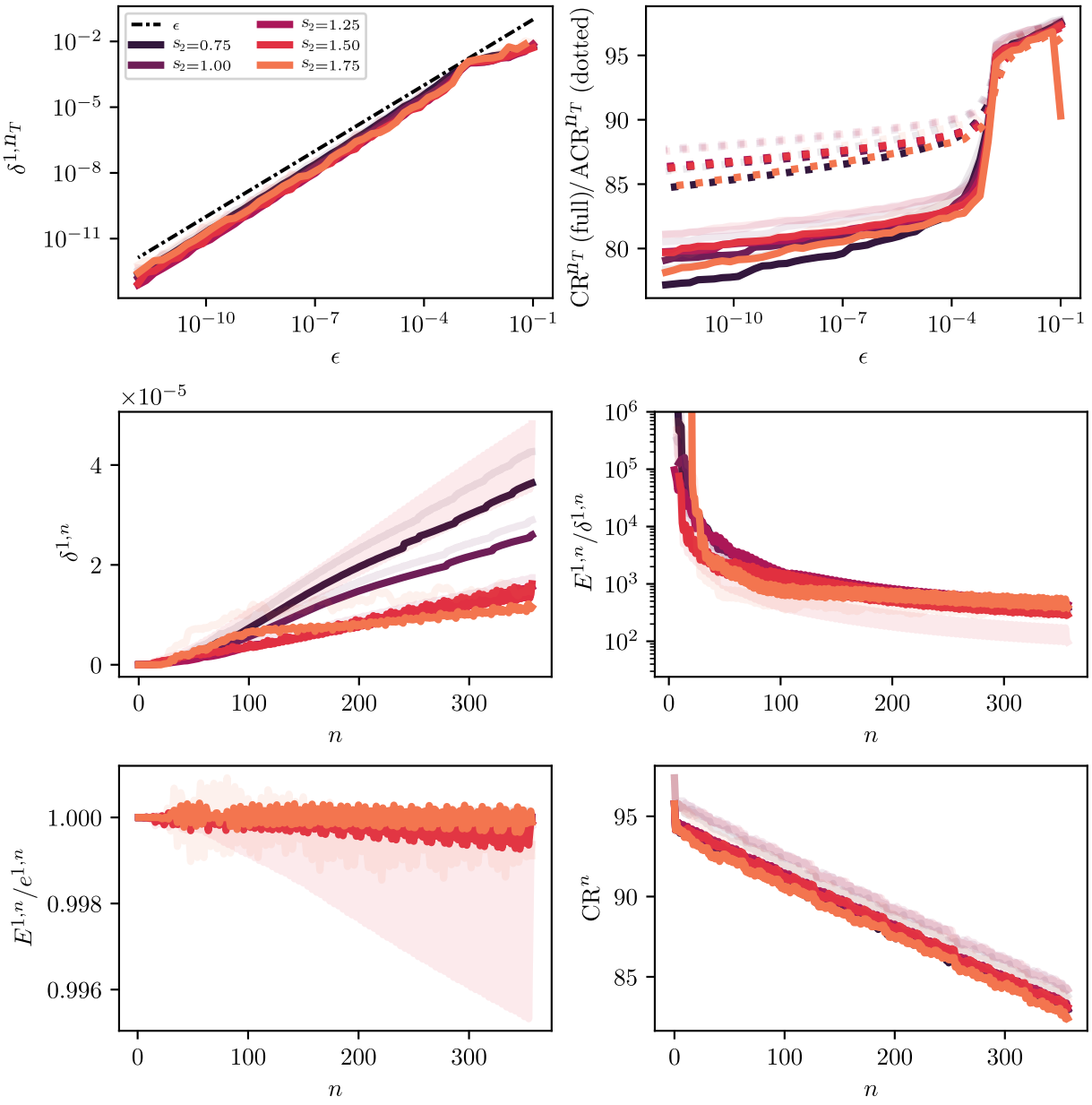


Figure 2.21: Test (IV) using $\gamma = 2$. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. The superimposed transparent lines refer to $\gamma = 1$ for comparison.

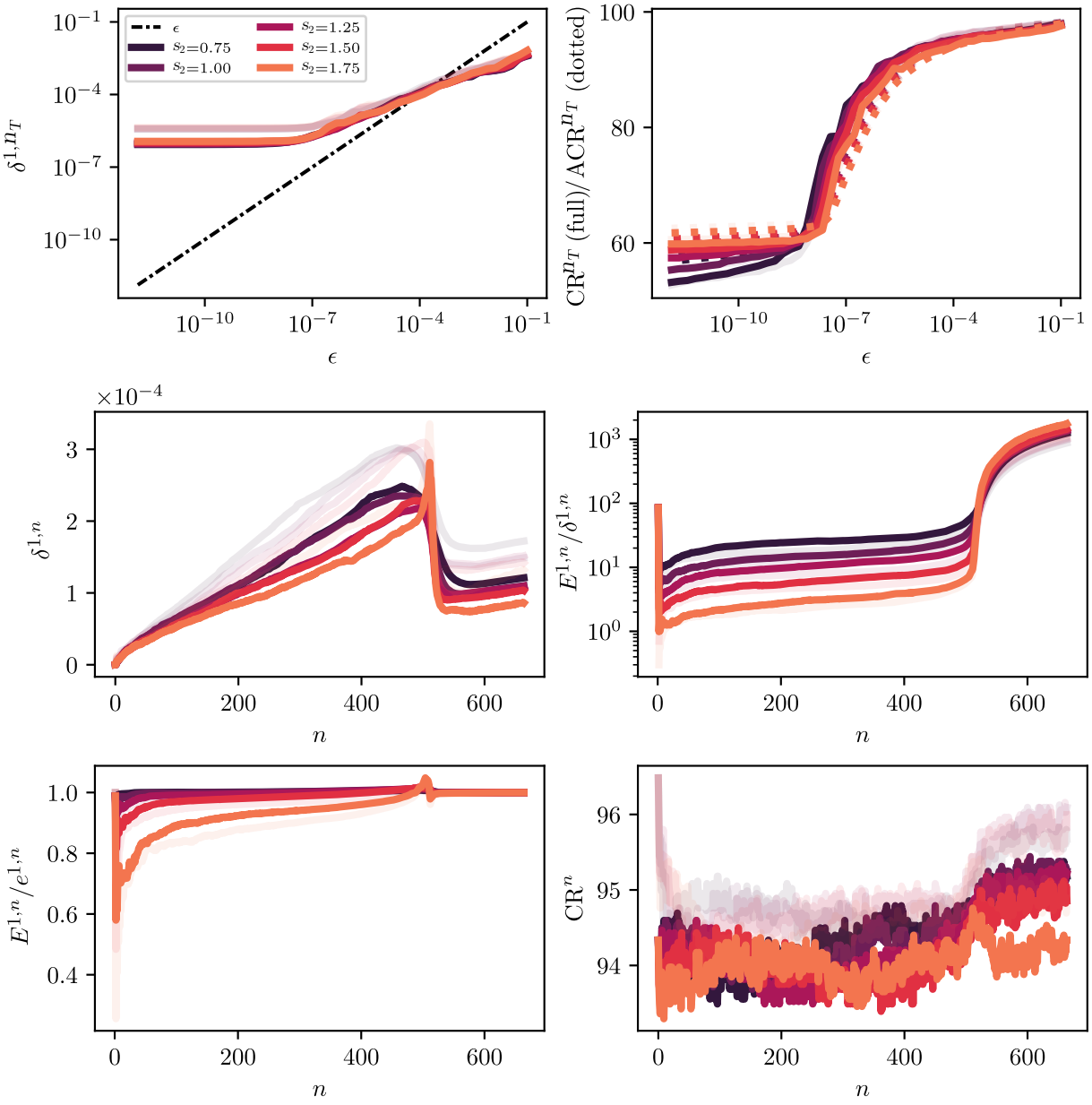


Figure 2.22: Test (V) using $\gamma = 2$. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and of $E^{1,n}/\delta^{1,n}$ (right) as functions of the time. Third row: behavior of $E^{1,n}/e^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. The superimposed transparent lines refer to $\gamma = 1$ for comparison.

We repeat the tests in order to provide the essential ideas about what happens when using multiresolution with a more precise prediction operator $\boldsymbol{P}_{\triangle}$, namely for $\gamma = 2$.

(I) Looking at Figure 2.18, we observe a smaller perturbation error δ^{1,n_T} compared to $\gamma = 1$, still following the right trend in ϵ and achieving better compression rates for the same values of ϵ. This is the direct consequence of having utilized a larger prediction stencil with a smooth solution, *cf.* Proposition 2.3.2. This is very inherent to the specific setting of smooth solutions. The time behavior of $\delta^{1,n}$ is linear with a smaller constant, which thus confirms to depend on γ, see Proposition 2.3.3 and Proposition 2.6.1.

(II) The performances shown in Figure 2.19 are comparable to those with $\gamma = 1$ because the solution lacks of smoothness: everything happens on the shock, where the details do not decay (*cf.* (2.26)) and therefore going from $\gamma = 1$ to $\gamma = 2$ produces essentially comparable outcomes. It is therefore not advisable to increase the cost of the overall procedure in this case.

(III) The results are in Figure 2.20 and the conclusions are essentially the same as (II), because the exact solution—though smooth—is very steep.

(IV) The results are in Figure 2.21 and the conclusions are essentially the same as (II).

(V) The results are in Figure 2.22 and the conclusions are essentially the same as (II). Moreover, we observe the stagnation of δ^{1,n_T} in ϵ both for $\gamma = 1$ and $\gamma = 2$, as previously studied.

Overall, the choice of $\gamma \geq 2$ is advisable, compared to $\gamma = 1$, only for very smooth problems, where it produces a significant gain. Otherwise, it does not yield major improvements of the quality of the solution and only generates more expensive computations due to the larger prediction stencil, as well as a more involved implementation and heavier meshes due to the grading constraint.

We can conclude that the adaptive scheme for a scalar conservation law guarantees an error control by a threshold ϵ and succeeds in keeping the perturbation error $\delta^{1,n}$ way smaller than the discretization error $E^{1,n}$ of the reference scheme, especially when weak solutions are involved, for the selected maximum level $\overline{\ell}$. The "leaves collision" does not impact these characteristics except in a specifically designed pathological case.

2.8.1.3 D_1Q_3 for the viscous Burgers equation: effect of the collision strategy

We now make a small break and further study the influence of several collision strategy on the outcome of the simulation. This will be the unique place in Section 2.8 where the computational mesh is not adapted in space and as time goes on. In order to see some difference between collision strategy, the equilibria must be non-linear.

2.8.1.3.1 The problem and the scheme We consider the approximation of the solution of the viscous Burgers equation with viscosity $\mu > 0$, given by

$$\begin{cases} \partial_t u + \partial_x(u^2) - \mu \partial_{xx} u = 0, & t \in [0,T], \quad x \in \mathbb{R}, \\ u(0,x) = \frac{1}{\sqrt{4\pi\mu t^\circ}} \exp\left(-\frac{x^2}{4\mu t^\circ}\right), & x \in \mathbb{R}, \end{cases}$$

where $t^\circ > 0$ is a parameter. The explicit solution is obtained by the Cole-Hopf transformation [Cole, 1951, Hopf, 1950] and is given by

$$u(t,x) = \sqrt{\frac{4\mu}{t}} \frac{\int_{-\infty}^{+\infty} \eta \exp\left(-\frac{1}{4\mu}\operatorname{erf}\left(\frac{x}{\sqrt{4\mu t^\circ}} - \sqrt{\frac{t}{t^\circ}}\eta\right)\right) e^{-\eta^2} \mathrm{d}\eta}{\int_{-\infty}^{+\infty} \exp\left(-\frac{1}{4\mu}\operatorname{erf}\left(\frac{x}{\sqrt{4\mu t^\circ}} - \sqrt{\frac{t}{t^\circ}}\eta\right)\right) e^{-\eta^2} \mathrm{d}\eta},$$

and the integrals with weights $e^{-\eta^2}$ shall be approximated with Gauss-Hermite formulæwith 100 quadrature points. We shall take either $\mu = 5\mathrm{e}-2$ (large diffusion) or $\mu = 5\mathrm{e}-3$ (small diffusion)

The numerical scheme that we use is the D_1Q_3 detailed in Section 1.5.2 using the choice of moments given by (1.5) and $N = 1$. We take the moments at equilibrium $m_2^{\mathrm{eq}}(\mathsf{m}_1) = (\mathsf{m}_1)^2/2$ and

$$m_3^{\mathrm{eq}}(\mathsf{m}_1) = \frac{(\mathsf{m}_1)^3}{3} + 4\mathsf{m}_1, \qquad s_2 = \left(\frac{1}{2} + \frac{\lambda\mu}{4\Delta x}\right)^{-1}, \qquad \text{(large diffusion)},$$

Table 2.4: Test for the viscous Burgers equation taking $\mu = 5e-2$ (small diffusion).

	Leaves collision (2.38)			Reconstructed collision (2.37)			Predict-and-integrate collision (2.39)		
$\Delta\underline{\ell}$	$E^{\underline{\ell}}_{\mathrm{coa}}$	$E^{\overline{\ell}}_{\mathrm{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\mathrm{coa}}$	$E^{\overline{\ell}}_{\mathrm{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\mathrm{coa}}$	$E^{\overline{\ell}}_{\mathrm{coa}}$	D_{coa}
0	1.23e-2	1.23e-2	0.00e+0	1.23e-2	1.23e-2	0.00e+0	1.23e-2	1.23e-2	5.18e-8
1	1.23e-2	1.23e-2	1.88e-7	1.23e-2	1.23e-2	1.14e-7	1.23e-2	1.23e-2	1.27e-7
2	1.23e-2	1.23e-2	9.34e-7	1.23e-2	1.23e-2	5.70e-7	1.23e-2	1.23e-2	5.76e-7
3	1.23e-2	1.23e-2	3.89e-6	1.23e-2	1.23e-2	2.40e-6	1.23e-2	1.23e-2	2.41e-6
4	1.23e-2	1.23e-2	1.57e-5	1.23e-2	1.23e-2	9.78e-6	1.23e-2	1.23e-2	9.79e-6
5	1.23e-2	1.23e-2	6.30e-5	1.23e-2	1.23e-2	4.06e-5	1.23e-2	1.23e-2	4.06e-5
6	1.23e-2	1.23e-2	2.60e-4	1.23e-2	1.23e-2	1.86e-4	1.23e-2	1.23e-2	1.86e-4
7	1.22e-2	1.22e-2	1.18e-3	1.22e-2	1.23e-2	9.97e-4	1.22e-2	1.23e-2	9.98e-4

Table 2.5: Test for the viscous Burgers equation taking $\mu = 5e-3$ (small diffusion).

	Leaves collision (2.38)			Reconstructed collision (2.37)			Predict-and-integrate collision (2.39)		
$\Delta\underline{\ell}$	$E^{\underline{\ell}}_{\mathrm{coa}}$	$E^{\overline{\ell}}_{\mathrm{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\mathrm{coa}}$	$E^{\overline{\ell}}_{\mathrm{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\mathrm{coa}}$	$E^{\overline{\ell}}_{\mathrm{coa}}$	D_{coa}
0	5.31e-3	5.31e-3	0.00e+0	5.31e-3	5.31e-3	0.00e+0	5.31e-3	5.31e-3	1.19e-6
1	5.31e-3	5.31e-3	3.47e-6	5.31e-3	5.31e-3	2.79e-6	5.31e-3	5.31e-3	3.02e-6
2	5.31e-3	5.31e-3	2.34e-5	5.31e-3	5.31e-3	2.28e-5	5.31e-3	5.31e-3	2.29e-5
3	5.28e-3	5.30e-3	1.41e-4	5.28e-3	5.28e-3	1.43e-4	5.28e-3	5.28e-3	1.43e-4
4	5.28e-3	5.31e-3	8.63e-4	5.29e-3	5.27e-3	8.93e-4	5.29e-3	5.27e-3	8.93e-4
5	5.92e-3	6.14e-3	6.08e-3	5.75e-3	5.83e-3	5.73e-3	5.75e-3	5.84e-3	5.76e-3
6	2.91e-2	3.36e-2	3.37e-2	2.67e-2	3.11e-2	3.14e-2	2.67e-2	3.12e-2	3.15e-2
7	2.55e-1	2.45e-1	2.42e-1	2.37e-1	2.27e-1	2.23e-1	2.32e-1	2.22e-1	2.19e-1

$$m_3^{\mathrm{eq}}(\mathsf{m}_1) = \frac{(\mathsf{m}_1)^3}{3} + \mathsf{m}_1, \qquad s_2 = \left(\frac{1}{2} + \frac{\lambda\mu}{\Delta x}\right)^{-1}, \qquad \text{(small diffusion)}.$$

We consider a final time $T = 1$, $t^\circ = 1$ and a domain $\Omega = [-3, 3]$. We take lattice velocity $\lambda = 4$ and $s_3 = 1$. According to the discussion by [Boghosian et al., 2018], the scheme is in general not convergent towards the solution of continuous equation, because $s_2 \to 0$ as $\Delta x \to 0$. Still, it can be used in an intermediate regime where Δx is not too small, thus s_2 is sufficiently away from 0.

2.8.1.3.2 Results As announced, for this test, we simulate over a uniform coarse mesh at level $\underline{\ell}$ (which will be changed) with a reference maximum level $\overline{\ell} = 11$, without performing any mesh adaptation. We monitor the following metrics on the conserved moment $\mathsf{m}_1 \approx u$, which are all taken with respect to the L^1 norm and normalized using the norm of the exact solution. They are considered at final time T.

- $E^{\underline{\ell}}_{\mathrm{coa}} = \|\overline{\boldsymbol{u}}(T) - \overline{\mathbf{m}}^{\mathrm{coa},1}(T)\|_{\ell^1} / \|\overline{\boldsymbol{u}}(T)\|_{\ell^1}$, error of the adaptive scheme applied on the uniform coarse mesh with respect to the exact solution measured at level $\underline{\ell}$.
- $E^{\overline{\ell}}_{\mathrm{coa}} = \|\overline{\boldsymbol{u}}(T) - \hat{\overline{\mathbf{m}}}^{\mathrm{coa},1}_{\overline{\ell}}(T)\|_{\ell^1} / \|\overline{\boldsymbol{u}}(T)\|_{\ell^1}$, error of the adaptive scheme applied on the uniform coarse mesh with respect to the exact solution measured at level $\overline{\ell}$, with the solution of the adaptive scheme built at $\overline{\ell}$ using the reconstruction operator.
- $D_{\mathrm{coa}} = \|\overline{\mathbf{m}}^{\mathrm{ref},1}(T) - \hat{\overline{\mathbf{m}}}^{\mathrm{coa},1}_{\overline{\ell}}(T)\|_{\ell^1} / \|\overline{\boldsymbol{u}}(T)\|_{\ell^1}$, difference between the reference and adaptive scheme applied on the uniform coarse mesh, where the adaptive datum has been reconstructed at finest level $\overline{\ell}$ in order to compare it with the solution of the reference scheme.

We test using the leaves collision (2.38), the reconstructed collision (2.37) and the predict-and-integrate collision (2.39). The latter is employed with a Gauss-Legendre quadrature of order five [Abramowitz and Stegun, 1964]. On the reference interval $[-1, 1]$, this corresponds to the quadrature points and weights $(x_1, w_1) = (-\sqrt{3/5}, 5/9)$, $(x_2, w_2) = (0, 8/9)$ and $(x_3, w_3) = (\sqrt{3/5}, 5/9)$.

The result in the case of large diffusion is given in Table 2.4. We observe that for this smooth solution, the additional error D_{coa} induced by the collision performed on the leaves is slightly larger (about 1.5 times) than those for the reconstructed collision and the predict-and-integrate collision. Still, all the errors have the same order of magnitude. On the other hand, the predict-and-integrate method behaves almost like the reconstructed method except for $\Delta\ell$ where it does not have $D_{\mathrm{coa}} = 0$ because the method does not perfectly coincide with the reference one. Regarding the case of small diffusion in Table 2.5, we obtain similar results, with the leaves collision showing marginally larger additional errors. This test shows that this strategy can be regarded as reliable even for

functions which do not behave polynomially, as the solution we considered, and which can present steep fronts. If one desires a slightly more qualitative collision strategy without significantly increase the computational cost, one may consider the predict-and-integrate strategy. Overall, the reconstructed collision, although guaranteeing the most accurate results, is generally not a viable choice due to its cost and the fact that it provides performances which are marginally better than the other cheaper strategies.

2.8.1.4 D_1Q_3/D_1Q_5 for two conservation laws

We come back to a setting where dynamically adaptive meshes are used. We consider two schemes. Indeed, the second one has an extended stencil, vising two neighboring cells for each direction. This tries to demonstrate that the method works well for very generic lattice Boltzmann schemes.

2.8.1.4.1 The problem and the scheme We consider the approximation of the weak entropic solution of the initial-value problem for the shallow water system

$$\begin{cases} \partial_t h + \partial_x(hu) = 0, & t \in [0,T], & x \in \mathbb{R}, \\ \partial_t(hu) + \partial_x(hu^2 + gh^2/2) = 0, & t \in [0,T], & x \in \mathbb{R}, \\ (h,u)(0,x) = (h^\circ, u^\circ)(x), & & x \in \mathbb{R}, \end{cases} \tag{2.51}$$

where h represents the height of a fluid and u its horizontal velocity. The parameter $g > 0$ is the gravitational acceleration exerted on the fluid and $h^\circ, u^\circ \in L^\infty(\mathbb{R})$.

We use two different lattice Boltzmann schemes with $N = 2$ conserved moments, under acoustic scaling:

- A D_1Q_3 introduced in Section 1.5.2 with moment matrix $\boldsymbol{M}$ given by (1.5). With the choice of moments at equilibrium
$$m_3^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2) = \frac{(\mathsf{m}_2)^2}{\mathsf{m}_1} + g\frac{(\mathsf{m}_1)^2}{2},$$
the theory of the equivalent equations [Dubois, 2008] allows to show that this scheme is first order consistent in Δx with (2.51), having $\mathsf{m}_1 \approx h$ and $\mathsf{m}_2 \approx hu$.
- A D_1Q_5 featuring $q = 5$ with $c_1 = 0$, $c_2 = 1$, $c_3 = -1$, $c_4 = 2$ and $c_5 = -2$ and moment matrix
$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & \lambda & -\lambda & 2\lambda & -2\lambda \\ 0 & \lambda^2 & \lambda^2 & 4\lambda^2 & 4\lambda^2 \\ 0 & \lambda^3 & -\lambda^3 & 8\lambda^3 & -8\lambda^3 \\ 0 & \lambda^4 & \lambda^4 & 16\lambda^4 & 16\lambda^4 \end{bmatrix}.$$
We select the moments at equilibrium as
$$m_3^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2) = \frac{(\mathsf{m}_2)^2}{\mathsf{m}_1} + g\frac{(\mathsf{m}_1)^2}{2}, \qquad m_4^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2) = \lambda^2 \mathsf{m}_1, \qquad m_5^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2) = \lambda^4 \mathsf{m}_1.$$
Again, this scheme is first order consistent in Δx with (2.51), having $\mathsf{m}_1 \approx h$ and $\mathsf{m}_2 \approx hu$.

As initial datum, we consider the Riemann problem given by

$$(h^\circ, u^\circ)(x) = (2,0)\mathbb{1}_{x<0}(x) + (1,0)\mathbb{1}_{x\geq 0}(x),$$

and the gravity to $g = 1$. The final time of the simulation is $T = 0.2$. We employ a computational domain $\Omega = [-1,1]$ and endow the schemes with oth order extrapolation boundary conditions. The lattice velocity is fixed to $\lambda = 2$ and we consider different relaxation parameters s_3, fixing $s_4 = s_5 = 1$ for the D_1Q_5 scheme.

2.8.1.4.2 Results We have:

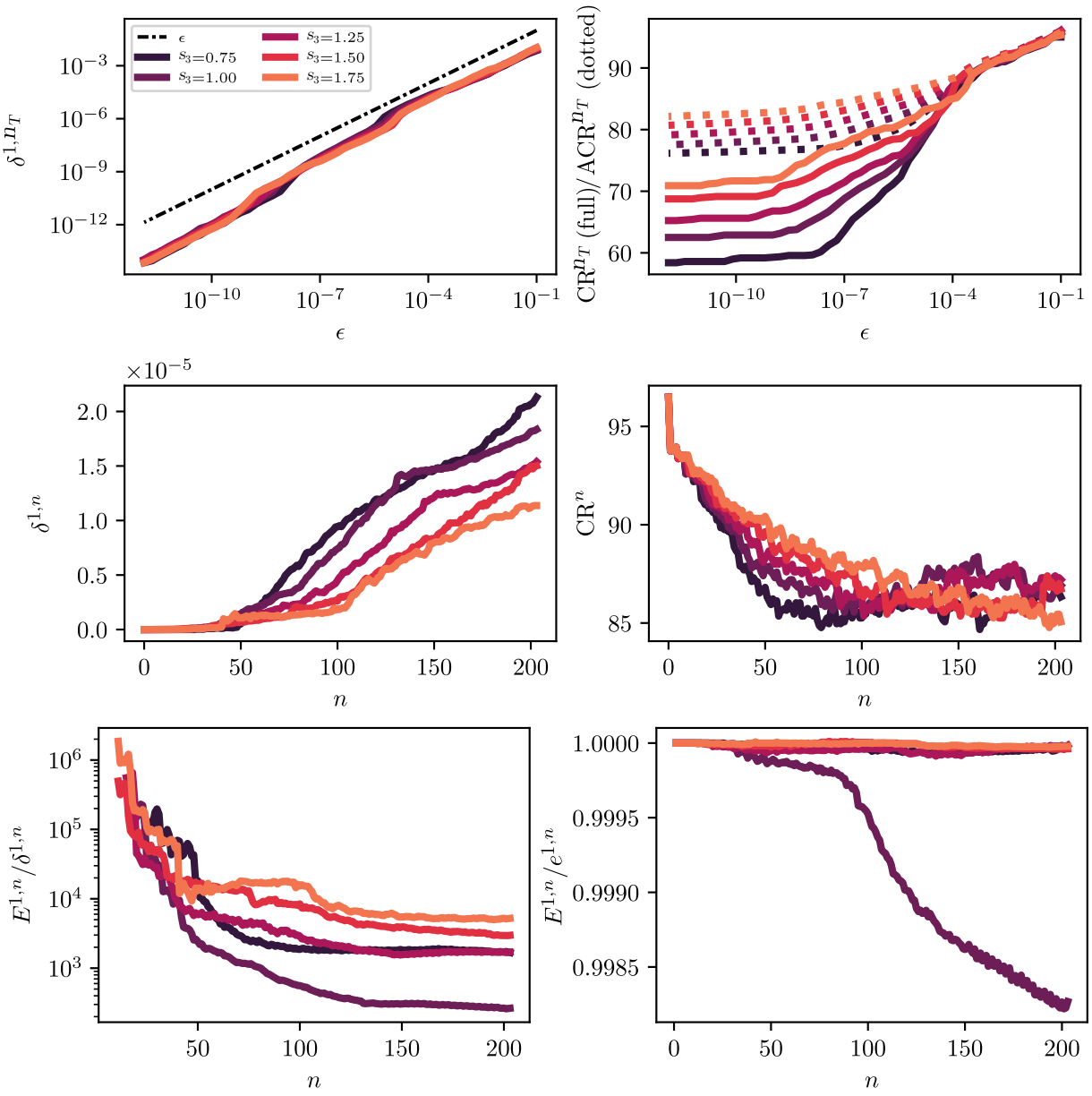


Figure 2.23: D_1Q_3 for the shallow-water system. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. Third row: behavior of $E^{1,n}/\delta^{1,n}$ (left) and $E^{1,n}/e^{1,n}$ (right) as functions of the time. For the sake of avoiding redundancy, the result are only for the first moment.

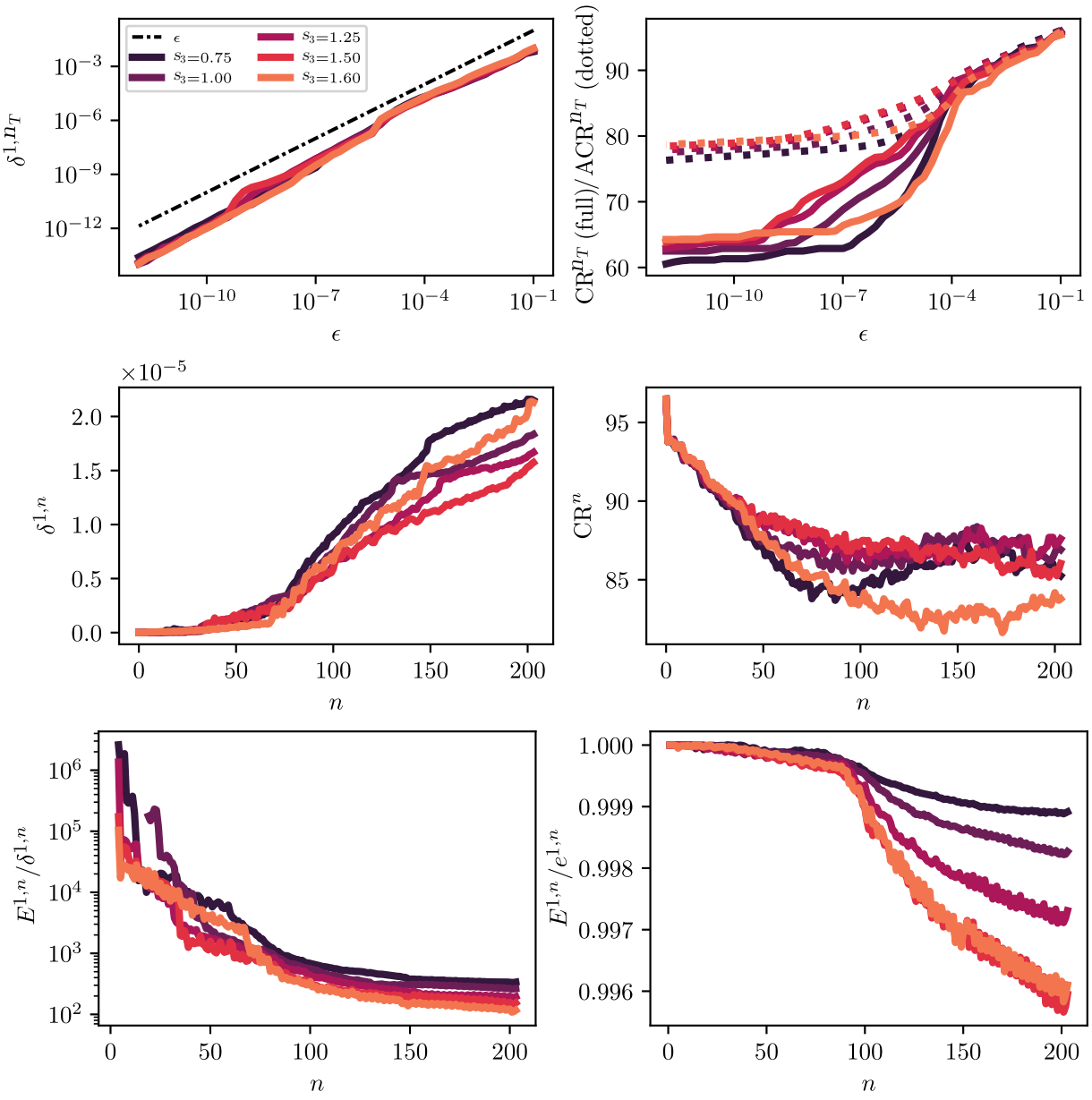


Figure 2.24: D_1Q_5 for the shallow-water system. First row: behavior of δ^{1,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{1,n}$ (left) and the compression rate CR^n (right) as functions of the time. Third row: behavior of $E^{1,n}/\delta^{1,n}$ (left) and $E^{1,n}/e^{1,n}$ (right) as functions of the time. For the sake of avoiding redundancy, the result are only for the first moment.

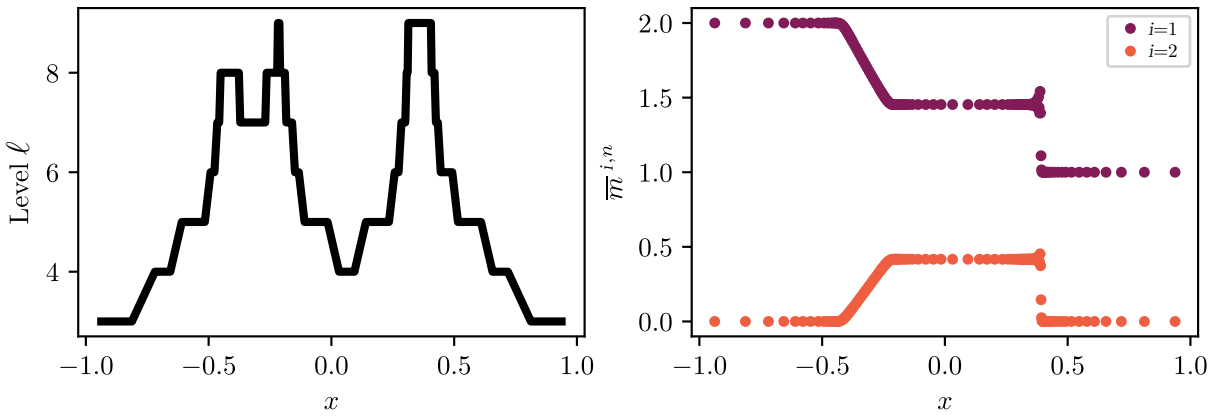


Figure 2.25: Example of solution from the D_1Q_5 adaptive scheme, considering $n = 300$, $s_2 = 1.6$ and $\epsilon = 1e-4$. On the left, levels of the computational mesh. On the right, solution on the leaves of the tree.

- For the D_1Q_3 scheme, the results are in Figure 2.23 concerning the first moment, for the sake of avoiding redundancy. For both the conserved moments, the behavior of the perturbation error in time is supralinear, being very small at the very beginning because the method adds enough security cells around the shock and information propagates relatively slowly. Moreover, we remark that the perturbation error is larger for smaller s_3 due to the larger diffusivity of the numerical scheme. The perturbation error is between four and six orders of magnitude smaller than the discretization error of the reference method, reaching very interesting compression factors. The estimates for δ^{i,n_T} for $i \in [\![1,2]\!]$ in terms of ϵ are correctly followed. We observe the typical inequality $\mathrm{ACR}^{n_T} > \mathrm{CR}^{n_T}$.
- For the D_1Q_5 scheme, the results are in Figure 2.24 and the behavior of the numerical solution is sketched in Figure 2.25. The time behavior of the perturbation error is again supralinear and now the difference between different relaxation parameters is less evident. The ratio with the discretization error of the reference scheme is between 10^4 and 10^6. The bound of δ^{i,n_T} for $i \in [\![1,2]\!]$ in ϵ is very well fulfilled. This example shows that our adaptive strategy works really well even for schemes with an extended advection stencil.

2.8.1.5 $D_1Q_2^3$ for three conservation laws

This test is constructed to check that our strategy works equally well with the so-called "vectorial schemes".

2.8.1.5.1 The problem and the scheme We consider the approximation of the weak entropic solution of the initial-value problem for the full Euler system

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, & t \in [0,T], & x \in \mathbb{R}, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0, & t \in [0,T], & x \in \mathbb{R}, \\ \partial_t E + \partial_x(Eu + pu) = 0, & t \in [0,T], & x \in \mathbb{R}, \\ (\rho, u, E)(0,x) = (\rho^\circ, u^\circ, E^\circ)(x), & & x \in \mathbb{R}, \end{cases} \tag{2.52}$$

where ρ is the density of the fluid, u is the velocity of the flow, p is the pressure and E the total energy. The pressure and the energy are linked by the pressure law which reads $E = \rho u^2/2 + p/(\gamma_{\mathrm{gas}} - 1)$, where γ_{gas} is the gas constant. The initial data are such that $\rho^\circ, u^\circ, E^\circ \in L^\infty(\mathbb{R})$.

As numerical scheme, we employ a vectorial scheme [Graille, 2014, Dubois, 2014] under acoustic scaling, which might be called $D_1Q_2^3$ and seen as a juxtaposition of three independent D_1Q_2 schemes with one conserved moment for each scheme, *cf.* Section 1.5.1, coupled *via* their equilibria. This scheme adds the necessary numerical diffusion, enhancing stability, and makes it easy to conserve the energy E with a used-defined pressure law. Another way of seeing this scheme is to consider a D_1Q_6 scheme with $q = 6$ and $N = 3$ with $c_1 = 1$, $c_2 = -1$, $c_3 = 1$, $c_4 = -1$, $c_5 = 1$

and $c_6 = -1$. The moment matrix is

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ \lambda & -\lambda & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda & -\lambda \end{bmatrix}.$$

Selecting the moments at equilibrium as

$$m_4^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2, \mathsf{m}_3) = \mathsf{m}_2, \qquad m_5^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2, \mathsf{m}_3) = \frac{1}{2}(3 - \gamma_{\mathrm{gas}}) \frac{(\mathsf{m}_2)^2}{\mathsf{m}_1} + (\gamma_{\mathrm{gas}} - 1)\mathsf{m}_3,$$
$$m_6^{\mathrm{eq}}(\mathsf{m}_1, \mathsf{m}_2, \mathsf{m}_3) = \gamma_{\mathrm{gas}} \frac{\mathsf{m}_2 \mathsf{m}_3}{\mathsf{m}_1} + \frac{1}{2}(1 - \gamma_{\mathrm{gas}}) \frac{(\mathsf{m}_2)^3}{(\mathsf{m}_1)^2},$$

[Dubois, 2008] allows to conclude [Graille, 2014] that this scheme is first order consistent in Δx with (2.52), having $\mathsf{m}_1 \approx \rho$, $\mathsf{m}_2 \approx \rho u$ and $\mathsf{m}_3 \approx E$.

As initial datum, we consider the Riemann problem associated with the well-known Sod shock problem [Sod, 1978], see [Toro, 2009, Chapter 4], given by

$$(\rho^\circ, u^\circ, E^\circ)(x) = (1.000, 0.000, 2.500)\mathbb{1}_{x<0}(x) + (0.125, 0.000, 0.250)\mathbb{1}_{x\geq 0}(x),$$

and we take $\gamma_{\mathrm{gas}} = 1.4$. The final time of the simulation is $T = 0.4$. We employ a computational domain $\Omega = [-1, 1]$ and endow all the schemes with oth order extrapolation boundary conditions. The lattice velocity is $\lambda = 3$. We take $s_4 = s_5 = s_6$, that is, the same relaxation parameter for each sub-scheme.

2.8.1.5.2 Results We monitor the same quantities as in Section 2.8.1.2. We have the results on Figure 2.26 and Figure 2.27: the perturbation error behaves fairly linearly in time for every choice of relaxation parameter and becomes smaller as s_3 approaches two, due to the reduced numerical diffusion. We are capable of keeping the perturbation error between three and four orders of magnitude smaller than the discretization error of the reference scheme for each of the conserved moments, for the chosen resolution $\overline{\ell}$. The behavior in ϵ is respected. This shows that our strategy is well suited to handle the simulation of systems of conservation laws using vectorial schemes.

2.8.2 2D and 3D tests

We test on 2D/3D problems, which start to look like real problems in terms of size and are more involved in terms of computational cost.

2.8.2.1 $\mathrm{D_2Q_4^3}$ for three conservation laws

2.8.2.1.1 The problem and the scheme We consider the approximation of the weak entropic solution of the initial-value problem for the full Euler system

$$\begin{cases} \partial_t \rho + \partial_{x_1}(\rho u) + \partial_{x_2}(\rho v) = 0, & t \in [0, T], & \boldsymbol{x} \in \mathbb{R}^2, \\ \partial_t(\rho u) + \partial_{x_1}(\rho u^2 + p) + \partial_{x_2}(\rho u v) = 0, & t \in [0, T], & \boldsymbol{x} \in \mathbb{R}^2, \\ \partial_t(\rho v) + \partial_{x_1}(\rho u v) + \partial_{x_2}(\rho v^2 + p) = 0, & t \in [0, T], & \boldsymbol{x} \in \mathbb{R}^2, \\ \partial_t E + \partial_{x_1}(E u + p u) + \partial_{x_2}(E v + p v) = 0, & t \in [0, T], & \boldsymbol{x} \in \mathbb{R}^2, \\ (\rho, u, v, E)(0, x) = (\rho^\circ, u^\circ, v^\circ, E^\circ)(x), & & \boldsymbol{x} \in \mathbb{R}^2, \end{cases} \tag{2.53}$$

where ρ is the density of the fluid, u is the velocity of the flow along the first axis and v along the second one, p is the pressure and E the total energy. The pressure and the energy are linked by the pressure law which reads

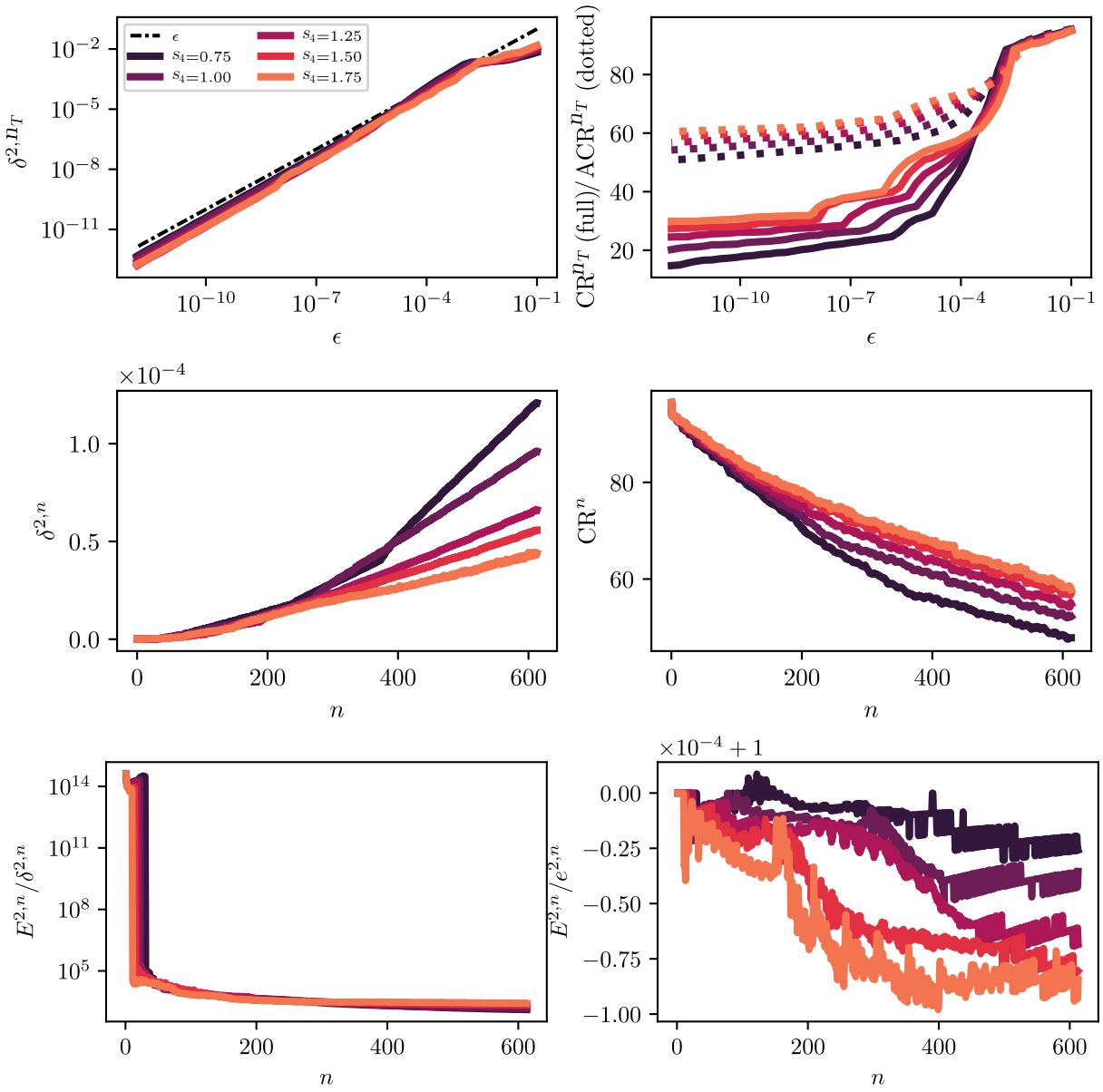


Figure 2.26: $D_1Q_2^3$ for the Euler system. First row: behavior of δ^{2,n_T} (left) and compression factors (right) at the final time T as functions of the threshold ϵ. Second row: behavior of $\delta^{2,n}$ (left) and the compression rate CR^n (right) as functions of the time. Third row: behavior of $E^{2,n}/\delta^{2,n}$ (left) and $E^{2,n}/e^{2,n}$ (right) as functions of the time. For the sake of avoiding redundancy, the result are only for the second moment.

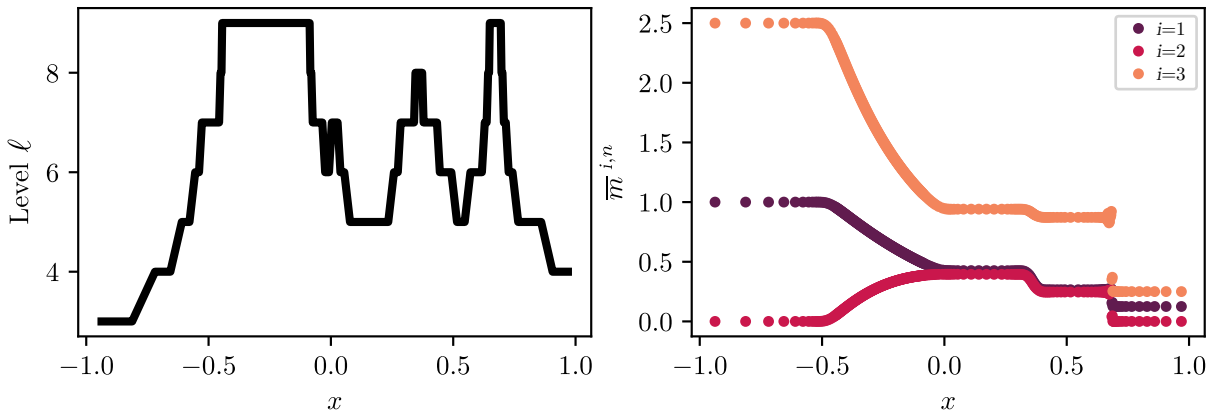


Figure 2.27: Example of solution from the $D_1Q_2^3$ adaptive scheme, considering $n = 600$, $s_3 = 1.75$ and $\epsilon = 1e-3$. On the left, levels of the computational mesh. On the right, solution on the leaves of the tree.

$E = \rho(u^2+v^2)/2 + p/(\gamma_{\text{gas}}-1)$, where γ_{gas} is the gas constant. The initial data are such that $\rho^\circ, u^\circ, v^\circ, E^\circ \in L^\infty(\mathbb{R})$.

As numerical scheme, we utilize a vectorial scheme under acoustic scaling, namely a $D_2Q_4^4$ made up of the juxtaposition of four D_2Q_4 with one conserved moment for each scheme, *cf.* Section 1.5.3, coupled through the equilibria. They can also be seen as a D_2Q_{16} scheme with $q = 16$ and

$$\boldsymbol{c}_j = \left(\cos\left(\frac{\pi}{2}((j-1) \bmod 4)\right), \sin\left(\frac{\pi}{2}((j-1) \bmod 4)\right)\right)^{\mathrm{t}}, \qquad j \in [\![1,16]\!],$$

where $a \bmod b$ is the remainder of the integer division between a and b. The moment matrix is

$$\boldsymbol{M} = \begin{bmatrix}
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
\lambda & 0 & -\lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \lambda & 0 & -\lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \lambda & 0 & -\lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \lambda & 0 & -\lambda & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda & 0 & -\lambda & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda & 0 & -\lambda & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda & 0 & -\lambda & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda & 0 & -\lambda \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2
\end{bmatrix}.$$

We recall that $m_i^{\text{eq}} \equiv m_i^{\text{eq}}(\mathsf{m}_1, \mathsf{m}_2, \mathsf{m}_3, \mathsf{m}_4)$ for $i \in [\![5,16]\!]$. Selecting

$$m_5^{\text{eq}} = \mathsf{m}_2, \qquad m_6^{\text{eq}} = \mathsf{m}_3, \qquad m_7^{\text{eq}} = 0,$$

$$m_8^{\text{eq}} = \frac{1}{2}(3-\gamma_{\text{gas}})\frac{(\mathsf{m}_2)^2}{\mathsf{m}_1} + \frac{1}{2}(1-\gamma_{\text{gas}})\frac{(\mathsf{m}_3)^2}{\mathsf{m}_1} + (\gamma_{\text{gas}}-1)\mathsf{m}_4, \qquad m_9^{\text{eq}} = \frac{\mathsf{m}_2\mathsf{m}_3}{\mathsf{m}_1}, \qquad m_{10}^{\text{eq}} = 0,$$

$$m_{11}^{\text{eq}} = \frac{\mathsf{m}_2\mathsf{m}_3}{\mathsf{m}_1}, \qquad m_{12}^{\text{eq}} = \frac{1}{2}(3-\gamma_{\text{gas}})\frac{(\mathsf{m}_3)^2}{\mathsf{m}_1} + \frac{1}{2}(1-\gamma_{\text{gas}})\frac{(\mathsf{m}_2)^2}{\mathsf{m}_1} + (\gamma_{\text{gas}}-1)\mathsf{m}_4, \qquad m_{13}^{\text{eq}} = 0,$$

$$m_{14}^{\text{eq}} = \gamma_{\text{gas}}\frac{\mathsf{m}_2\mathsf{m}_4}{\mathsf{m}_1} + \frac{1}{2}(1-\gamma_{\text{gas}})\left(\frac{(\mathsf{m}_2)^3}{(\mathsf{m}_1)^2} + \frac{\mathsf{m}_2(\mathsf{m}_3)^2}{(\mathsf{m}_1)^2}\right), \qquad m_{15}^{\text{eq}} = \gamma_{\text{gas}}\frac{\mathsf{m}_3\mathsf{m}_4}{\mathsf{m}_1} + \frac{1}{2}(1-\gamma_{\text{gas}})\left(\frac{(\mathsf{m}_3)^3}{(\mathsf{m}_1)^2} + \frac{\mathsf{m}_3(\mathsf{m}_2)^2}{(\mathsf{m}_1)^2}\right),$$

and $m_{16}^{\mathrm{eq}} = 0$, the scheme [Dubois, 2008] is first order consistent in Δx with (2.53), having $\mathsf{m}_1 \approx \rho$, $\mathsf{m}_2 \approx \rho u$, $\mathsf{m}_3 \approx \rho v$ and $\mathsf{m}_4 \approx E$.

As initial datum, we take the Configuration 3 and 12 by [Lax and Liu, 1998], under the form

$$(\rho^\circ, u^\circ, v^\circ, E^\circ)(x) = \begin{cases} (\rho^\circ_{\mathrm{UR}}, u^\circ_{\mathrm{UR}}, v^\circ_{\mathrm{UR}}, E^\circ_{\mathrm{UR}}), & x_1 > 1/2, \quad x_2 > 1/2, \\ (\rho^\circ_{\mathrm{UL}}, u^\circ_{\mathrm{UL}}, v^\circ_{\mathrm{UL}}, E^\circ_{\mathrm{UL}}), & x_1 < 1/2, \quad x_2 > 1/2, \\ (\rho^\circ_{\mathrm{LL}}, u^\circ_{\mathrm{LL}}, v^\circ_{\mathrm{LL}}, E^\circ_{\mathrm{LL}}), & x_1 < 1/2, \quad x_2 < 1/2, \\ (\rho^\circ_{\mathrm{LR}}, u^\circ_{\mathrm{LR}}, v^\circ_{\mathrm{LR}}, E^\circ_{\mathrm{LR}}), & x_1 > 1/2, \quad x_2 < 1/2. \end{cases}$$

Moreover, we utilize $\gamma_{\mathrm{gas}} = 1.4$. The final time of the simulation is $T = 0.3$ for Configuration 3 and $T = 0.25$ for Configuration 12. We employ the computational domain $\Omega = [0,1]^2$ and endow all the schemes with oth order extrapolation conditions at the boundary. For the examined configurations, we found that $\lambda = 5$ and $s_5 = s_6 = 1.9$, $s_7 = s_{10} = s_{13} = s_{16} = 1$ and $s_8 = s_9 = s_{11} = s_{12} = s_{14} = s_{15} = 1.75$ provide adequate performances and a reasonable amount of numerical diffusion to keep simulations stable. We have also utilized the twisted scheme presented in Section 1.5.3, see [Février, 2014, Chapter 1] obtaining similar behavior. We do not present such tests.

2.8.2.1.2 *Results*

- **General remarks**. The structure of the solution and the local relative perturbation error concerning the density field (*i.e.* m_1) at final time for Configuration 3 are given in Figure 2.28; those for Configuration 12 are given in Figure 2.29. In the former case, we remark that the four shocks, where all the conserved moments are discontinuous, are well resolved and finely meshed, so that we can observe some hydrodynamic instabilities [Liska and Wendroff, 2003] typical of such systems—despite the fact of using a low order scheme. In the latter case, the two shocks propagating towards the upper-right corner are followed by the finest discretization $\overline{\ell}$ of the mesh, whereas we observe a coarsening of one level (for $\epsilon = 5\mathrm{e}-3$) close to the static contact discontinuities. This phenomenon shall be clarified in a moment with a finer analysis. Overall, this qualitative analysis allows us to conclude that the adaptive lattice Boltzmann scheme succeeds in reproducing the expected behavior of the solution [Lax and Liu, 1998, Liska and Wendroff, 2003] of the Euler system and that the adaptive grid follows the shock structures propagating with finite velocity.

- **Error control**. The first important point is to verify that we control the perturbation error. We monitor the relative perturbation error at final time T

$$E^i(T) = \frac{\delta^{i,n_T}}{\|\overline{\mathbf{m}}^{i,\mathrm{ref}}(T)\|_{\ell^1}},$$

with $i \in [\![1, N]\!]$ spanning the conserved moments and the normalization is performed using the reference solution. The notation δ^{i,n_T} is the one introduced in (2.49). According to Proposition 2.6.1, we would like to observe an error control of the form

$$E^i(T) \lesssim \epsilon, \tag{2.54}$$

where the constant depends on the final time T and type of multiresolution γ that one utilizes. Observe that as we are computing the difference with the reference scheme, the solution of the latter depends on the maximum level of resolution $\overline{\ell}$. This should be taken into account when comparing results for different $\overline{\ell}$.

According to the results shown in Figure 2.30 and Figure 2.31 (we present only for $i = 1$), respectively for Configuration 3 and 12, we verify that the upper bound (2.54) is verified for these test cases. This is in agreement with the theoretical analysis Proposition 2.6.1 and holds for any choice of finest resolution $\overline{\ell}$. Besides corroborating the theoretical analysis in a multidimensional setting, this confirms that, as far as we are dealing with hyperbolic problems, the enlargement strategy devised in Section 2.4 is capable of ensuring that the adaptive mesh correctly follows the temporal evolution of the solution.

- **Memory occupation**. The second important axis of analysis is the gain in terms of computational time and memory impact thanks to the joint work of the adaptive scheme with multiresolution. The memory

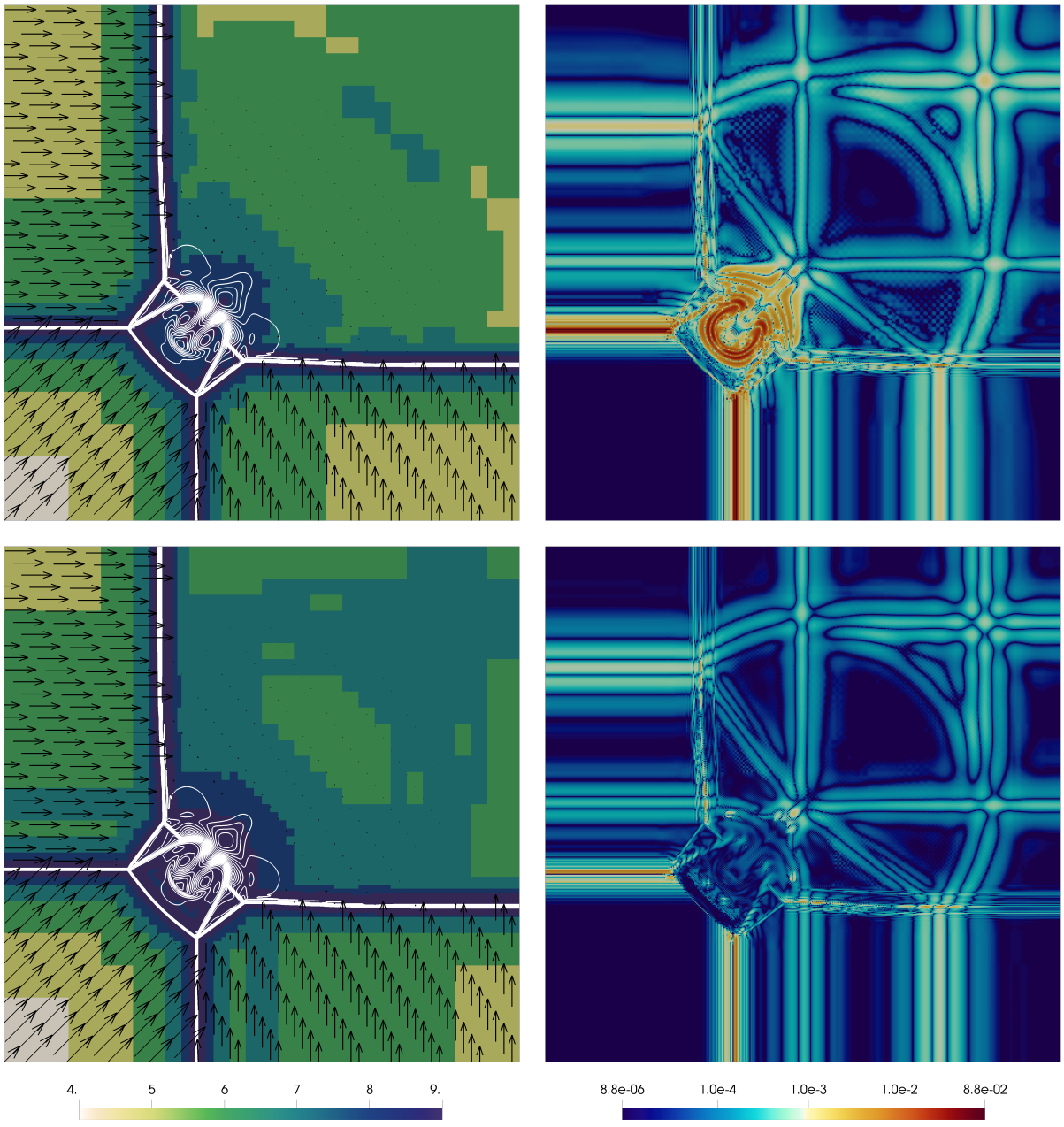


Figure 2.28: Configuration 3. Top $\epsilon = 5e-3$, bottom $\epsilon = 1e-3$ and $\overline{\mu} = 0$ (*cf.* (2.36)) for both. On the left, level ℓ (colored), contours of the density field m_1 (white) and velocity field $(m_2, m_3)^t$ (black). On the right: local relative perturbation error on the density field m_1 of the adaptive method (reconstructed solution) with respect to the reference method. Time $T = 0.3$, $\underline{\ell} = 2$ and $\overline{\ell} = 9$.

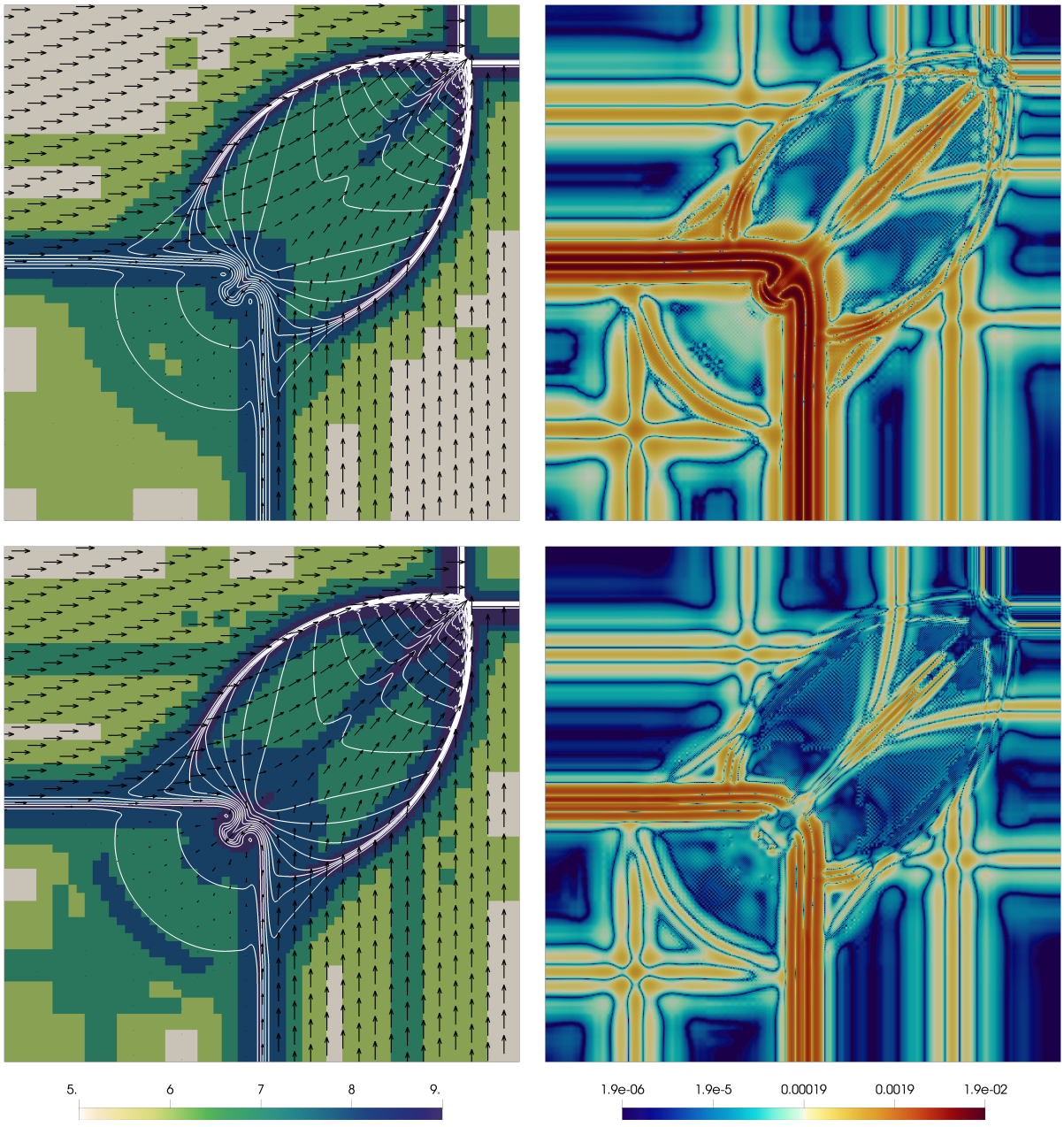


Figure 2.29: Configuration 12. Top $\epsilon = 5e-3$, bottom $\epsilon = 1e-3$ and $\overline{\mu} = 0$ (*cf.* (2.36)) for both. On the left, level ℓ (colored), contours of the density field m_1 (white) and velocity field $(m_2, m_3)^t$ (black). On the right: local relative perturbation error on the density field m_1 of the adaptive method (reconstructed solution) with respect to the reference method. Time $T = 0.25$, $\underline{\ell} = 2$ and $\overline{\ell} = 9$.

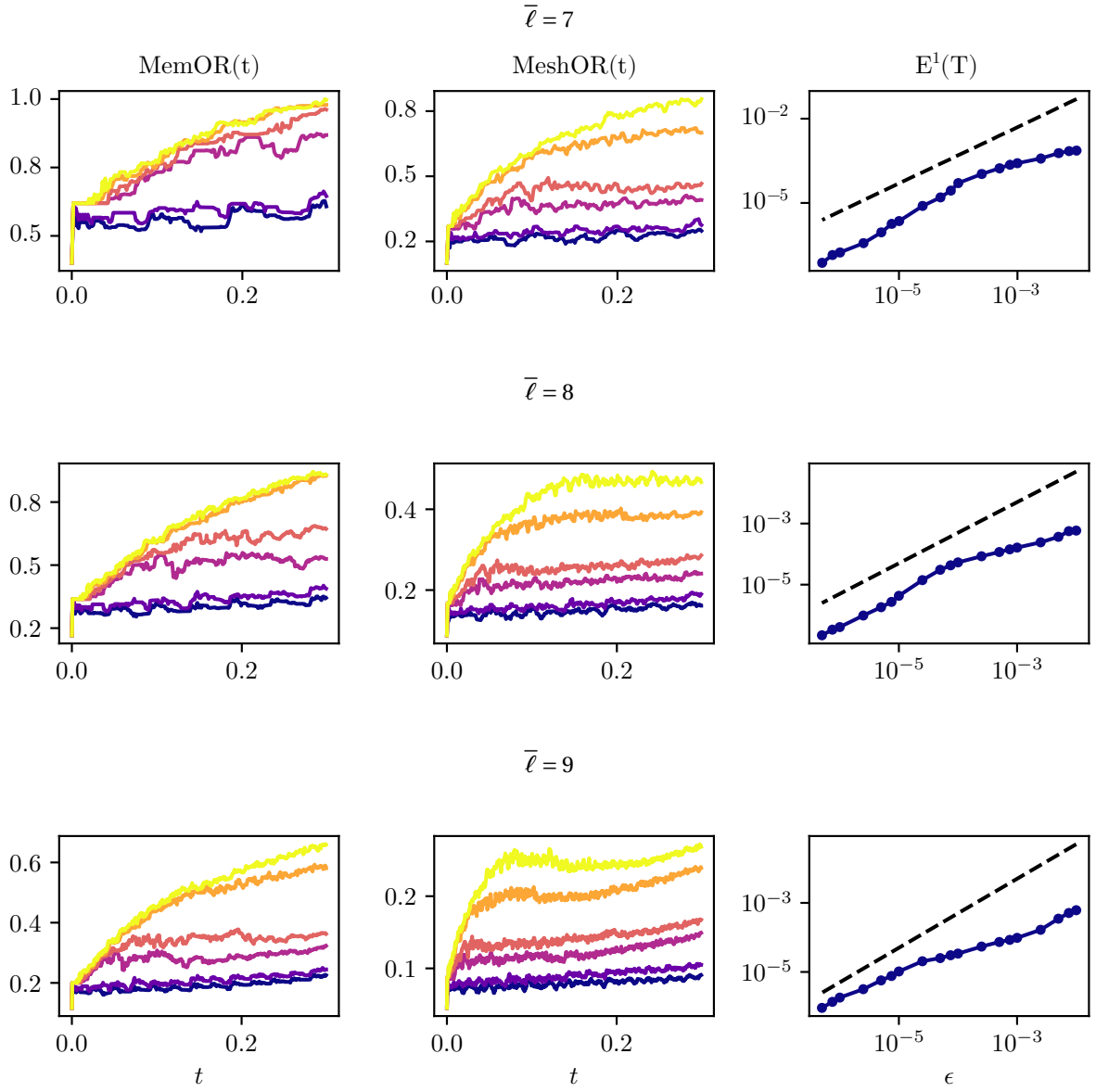


Figure 2.30: Configuration 3 with $\underline{\ell} = 2$ and $\overline{\mu} = 0$ (*cf.* (2.36)). From top to bottom $\overline{\ell} = 7, 8$ and 9. For the two plots on the left: $\epsilon = 1e-2$, $\epsilon = 5e-3$, $\epsilon = 1e-3$, $\epsilon = 5e-4$, $\epsilon = 1e-4$, $\epsilon = 5e-5$. The dashed black line gives the slope ϵ.

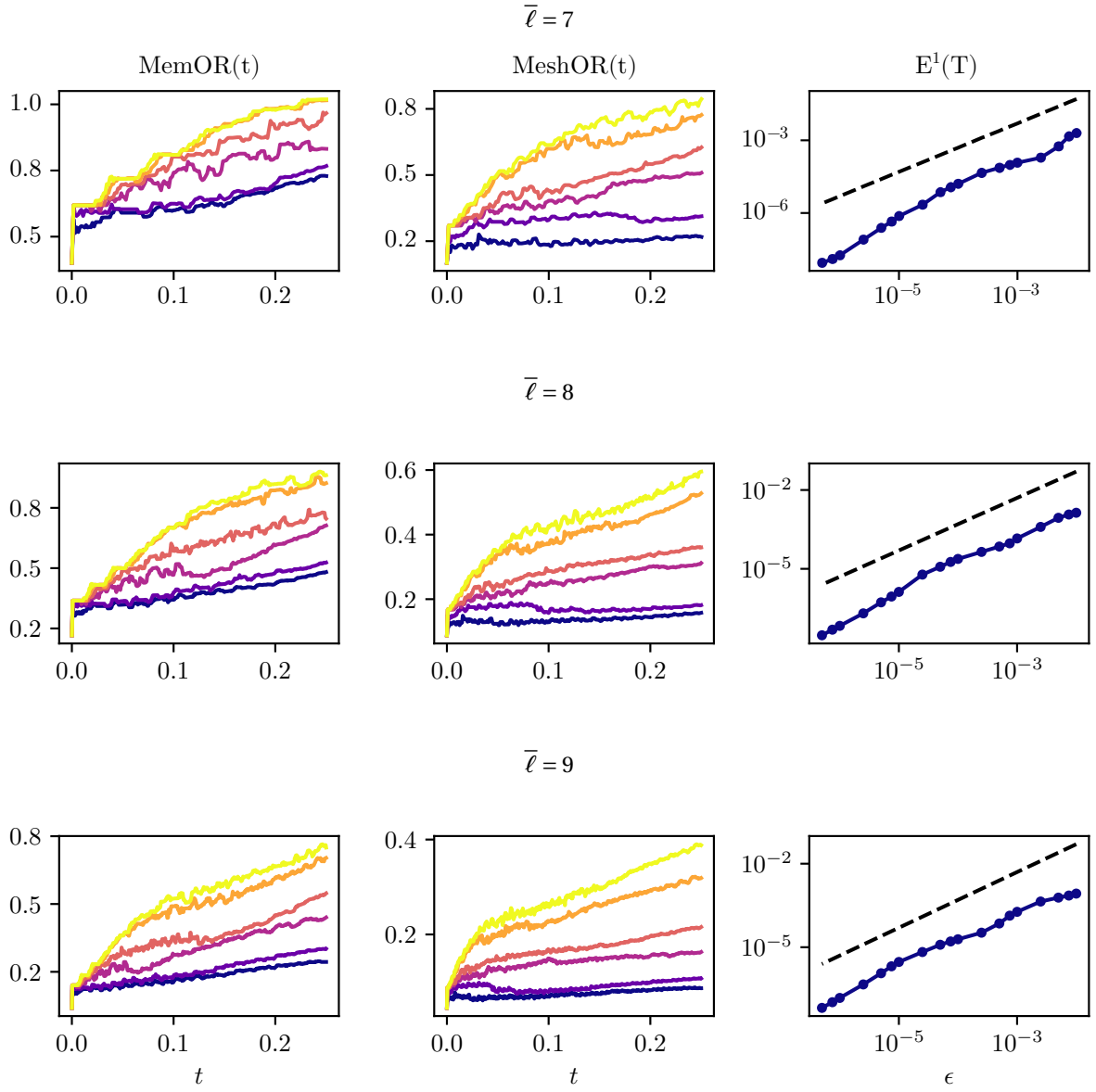


Figure 2.31: Configuration 12 with $\underline{\ell} = 2$ and $\overline{\mu} = 0$ (*cf.* (2.36)). From top to bottom $\overline{\ell} = 7, 8$ and 9. For the two plots on the left: $\epsilon = 1e-2$, $\epsilon = 5e-3$, $\epsilon = 1e-3$, $\epsilon = 5e-4$, $\epsilon = 1e-4$, $\epsilon = 5e-5$. The dashed black line gives the slope ϵ.

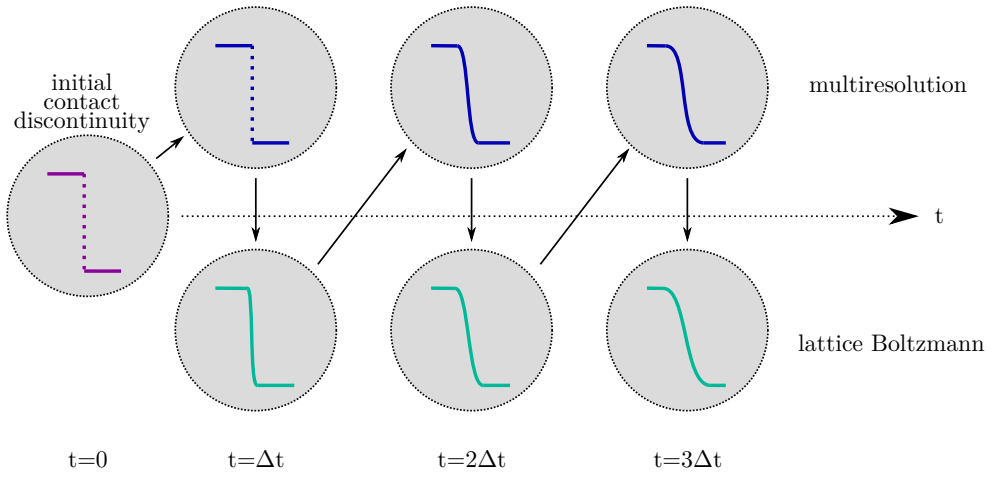


Figure 2.32: Coupling between the lattice Boltzmann scheme and the multiresolution to amplify the errors on the contact discontinuities.

occupation rate and the mesh occupation rate at time $t = n\Delta t$ are given by

$$\mathrm{MemOR}(t) = \frac{\#(\text{total cells to encode } \Lambda(t) \text{ in the data structure})}{\#(\text{total cells to encode the uniform mesh at level } \overline{\ell} \text{ in the data structure})},$$
$$\mathrm{MeshOR}(t) = \frac{\#(S(\Lambda(t)))}{N_{\overline{\ell}}} = 1 - \frac{\mathrm{CR}^n}{100}.$$

Observe that $\mathrm{MemOR}(t)$ depends on our choice of implementation, see Part II. Occupation rates much smaller than one are good and correspond to high compression rates.

Still looking at Figure 2.30 and Figure 2.31, we observe that the occupation rates, namely $\mathrm{MemOR}(t)$ and $\mathrm{MeshOR}(t)$ become more interesting as one approaches larger maximum levels of resolution $\overline{\ell}$. This is in accordance with the intuition that since a fine sampling of the solution is needed only close to the shocks, the number of needed leaves shall grow in $\overline{\ell}$ more slowly than $N_{\overline{\ell}}^d = 2^{d\overline{\ell}}$, also because shocks are $(d-1)$-dimensional entities.

As far as time is concerned, after an important initial growth guided by the refinement criterion $\mathcal{H}_\epsilon$, the trend of the occupation rates eventually stabilizes, especially for Configuration 3, where the secondary structures close to the hydrodynamic instabilities induced by the contact discontinuities do not grow too much in size as time advances. For Configuration 12, the occupation rates grow linearly even close to the final time because of the expanding curvilinear front linking the main shocks and the contact discontinuities.

Finally, the occupation rates are less interesting once we decrease ϵ. In these tests, we deliberately used unrealistically small ϵ only aiming at showing convergence. We conclude that the choice of ϵ should be the result of an arbitration between a desired target error and performances.

- **Coupling between multiresolution and lattice Boltzmann: the role of $\overline{\mu}$ in** (2.36). As it has been hinted while discussing Figure 2.29, we remark that the larger errors for Configuration 12 are situated close to the static contact discontinuities. This also holds, but less spectacularly, for Configuration 3 in the area where contact discontinuities are present causing the hydrodynamic instabilities to appear. This is an interesting coupling phenomenon between the poor behavior of the reference scheme on the contact discontinuities—which is inherent to this class of vectorial schemes [Graille, 2014]—and multiresolution. Indeed, the reference scheme smears the contact discontinuities (decreases the magnitude of the details in these areas) from the very beginning and then the multiresolution adaptation coarsens the mesh causing a local accumulation of error in time. This pattern is schematized in Figure 2.32.

  We also verified that the fact of performing the collision on the complete leaves (2.38) without reconstruction (2.37), has a negligible impact on this particular phenomenon, even if the equilibrium functions are strongly non-linear.

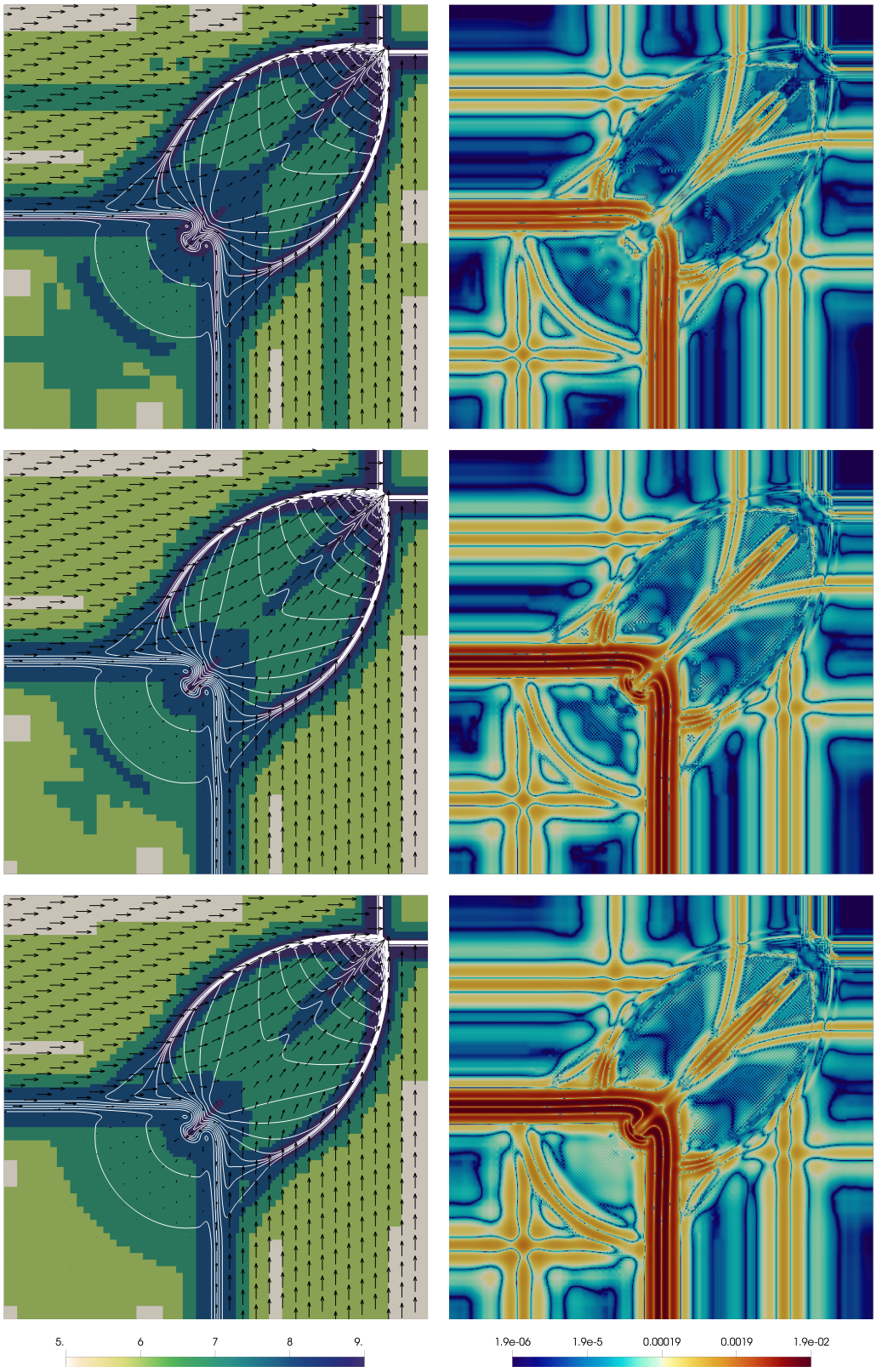


Figure 2.33: Configuration 12. For all the simulations, time $T = 0.25$, $\underline{\ell} = 2$ and $\overline{\ell} = 9$, $\epsilon = 0.001$. On the left, level ℓ (colored), contours of the density field (white) and velocity field (black). On the right: local relative perturbation error of the adaptive method with respect to the reference method. From top to bottom $\overline{\mu} = 0, 1$ and 2.

As Figure 2.29 and Figure 2.28 show, this problem is clearly alleviated by decreasing the threshold ϵ, from 5e−3 to 1e−3. Still, it is more interesting to study how the regularity guess $\overline{\mu}$ for the solution used in the refinement criterion (2.36) influences the behavior of the adaptive scheme. To study this for Configuration 12, we fixed $\epsilon = 1\mathrm{e}-3$ and varied $\overline{\mu} = 0, 1$ and 2, as shown in Figure 2.33. We observe that this parameter, involved in the refinement process $\mathcal{H}_\epsilon$, does not affect the structures which are already well refined after the coarsening process $\mathcal{T}_\epsilon$, namely the shocks. Close to these structures, if we assume that (2.26) is sharp, the details do not decrease with $\overline{\ell}$ and thus precision is ensured if ϵ is reasonably small. On the other hand, we observe that a wise choice of $\overline{\mu}$ is effective in diminishing the coupling effect between the multiresolution and the lattice Boltzmann scheme in smearing contact discontinuities (consider that the color-scale is logarithmic), without having to drastically reduce ϵ, which would cause a degradation of the performance of the algorithm. Since $\overline{\mu}$ represents the number of bounded derivatives of the expected solution of the problem, the advice for solutions developing shocks and contact discontinuities (thus only L^∞ solutions) is to set $\overline{\mu} = 0$. This has proved to allow for a reduction of the artificial smearing of the contact discontinuities by the numerical method.

2.8.2.2 D_2Q_9 for three conservation laws

This is the most paradigmatic application of lattice Boltzmann.

2.8.2.2.1 The problem and the scheme We consider the problem of a viscous flow around an obstacle occupying the open set $\Theta \subset \mathbb{R}^2$, with flow supposed to be incompressible and being a Newtonian fluid:

$$\begin{cases} \nabla_{\boldsymbol{x}} \cdot \boldsymbol{u} = 0, & t \in [0,T], & \boldsymbol{x} \in \mathbb{R}^2 \setminus \Theta, \\ \rho_0(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla_{\boldsymbol{x}} \boldsymbol{u}) + \nabla_{\boldsymbol{x}} p - \nabla_{\boldsymbol{x}} \cdot \left(2\mu \dfrac{\nabla_{\boldsymbol{x}} \boldsymbol{u} + \nabla_{\boldsymbol{x}} \boldsymbol{u}^{\mathrm{t}}}{2}\right) = 0, & t \in [0,T], & \boldsymbol{x} \in \mathbb{R}^2 \setminus \Theta, \\ \boldsymbol{u}(0,\boldsymbol{x}) = (u^\circ, 0)^{\mathrm{t}}, & & \boldsymbol{x} \in \mathbb{R}^2 \setminus \Theta, \\ \boldsymbol{u}(t,\boldsymbol{x}) = \boldsymbol{0}, & t \in [0,T], & \boldsymbol{x} \in \partial\Theta, \end{cases} \tag{2.55}$$

where $\boldsymbol{u} = (u, v)^{\mathrm{t}}$ is the velocity of the flow, ρ_0 the constant density, p is the Lagrange multiplier enforcing incompressibility and $\mu > 0$ is the dynamic viscosity. The initial velocity is along the first axis and equals u°. The leading dimensionless quantity in this problem, determining the flow regime, is the Reynolds number

$$\mathrm{Re} = \frac{\rho_0 u^\circ \sqrt{\int_\Theta \mathrm{d}\boldsymbol{x}}}{\mu}.$$

It is well-known that for $\mathrm{Re} > 90$, the flow passes from being fully laminar to a periodic regime where vortices periodically shed. This phenomenon is known as *von Kármán* vortex street.

As numerical scheme, we use the D_2Q_9 introduced in Section 1.5.4, under acoustic scaling, were we take $N = 3$ and moments at equilibrium

$$\begin{aligned} m_4^{\mathrm{eq}} &= -2\lambda^2 \mathsf{m}_1 + \frac{3((\mathsf{m}_2)^2 + (\mathsf{m}_3)^2)}{\mathsf{m}_1}, & m_5^{\mathrm{eq}} &= -\lambda^2 \mathsf{m}_2, & m_6^{\mathrm{eq}} &= -\lambda^2 \mathsf{m}_3, \\ m_7^{\mathrm{eq}} &= \lambda^4 \mathsf{m}_1 - \frac{3\lambda^2((\mathsf{m}_2)^2 + (\mathsf{m}_3)^2)}{\mathsf{m}_1}, & m_8^{\mathrm{eq}} &= \frac{(\mathsf{m}_2)^2 - (\mathsf{m}_3)^2}{\mathsf{m}_1}, & m_9^{\mathrm{eq}} &= \frac{\mathsf{m}_2 \mathsf{m}_3}{\mathsf{m}_1}. \end{aligned} \tag{2.56}$$

The relaxation parameters are taken as $s_4 = s_5 = s_6 = s_7$ and $s_8 = s_9$ Assuming to be in the low-Mach setting $|u^\circ|/\lambda \ll 1$, the equivalent equation analysis [Dubois, 2008], see [Février, 2014], gives consistency—upon neglecting terms proportional to the cube of the velocity field—with the following system

$$\begin{cases} \partial_t \rho + \nabla_{\boldsymbol{x}}(\rho \boldsymbol{u}) = O(\Delta x^2), \\ \partial_t(\rho \boldsymbol{u}) + \nabla_{\boldsymbol{x}} \cdot (\rho \boldsymbol{u} \otimes \boldsymbol{u}) + \nabla_{\boldsymbol{x}}\left(\dfrac{\lambda^2}{3}\rho\right) - \dfrac{\lambda \Delta x}{3} \nabla_{\boldsymbol{x}} \cdot \left(2\rho\left(\dfrac{1}{s_8} - \dfrac{1}{2}\right)\left(\dfrac{\nabla_{\boldsymbol{x}} \boldsymbol{u} + \nabla_{\boldsymbol{x}} \boldsymbol{u}^{\mathrm{t}}}{2}\right) + \rho\left(\dfrac{1}{s_4} - \dfrac{1}{s_8}\right)(\nabla_{\boldsymbol{x}} \cdot \boldsymbol{u})\boldsymbol{I}\right) = O(\Delta x^2), \end{cases} \tag{2.57}$$

being in a quasi-incompressible regime where $\mathsf{m}_1 \approx \rho \approx \rho_0$, $\mathsf{m}_2 \approx \rho u$ and $\mathsf{m}_3 \approx \rho v$. In order to enforce the viscosity by (2.55), one takes

$$s_8 = \left(\frac{1}{2} + \frac{3\mu}{\lambda \rho_0 \Delta x}\right)^{-1}.$$

Notice that the dissipation term is modeled using what is indeed numerical viscosity, being proportional to Δx and which is thus asymptotically vanishing. Special care must be devoted to the representation of the obstacle Θ. Usually, one employs bounce back boundary condition [Dubois et al., 2015] to enforce zero velocity on $\partial\Theta$ and there exist a vast specialized literature on this matter. However, in this work, we do not want to adapt the domain and its discretization to fit $\partial\Theta$ and so we proceed in the following way: we mesh the entire space $\Omega = [0,2] \times [0,1]$ without considering the obstacle and we apply the adaptive scheme as always. At the end of each time step, for the leaves $C_{\ell,\boldsymbol{k}}$ intersecting Θ, we estimate $|C_{\ell,\boldsymbol{k}} \cap \Theta|_d$ and we post-treat using

$$\bar{\mathbf{f}}_{\ell,\boldsymbol{k}}(t+\Delta t) = \frac{|C_{\ell,\boldsymbol{k}} \cap \Theta|_d}{|C_{\ell,\boldsymbol{k}}|_d} \boldsymbol{M}^{-1} \boldsymbol{m}^{\mathrm{eq}}(\mathsf{m}_1 = \rho_0, \mathsf{m}_2 = 0, \mathsf{m}_3 = 0) + \left(1 - \frac{|C_{\ell,\boldsymbol{k}} \cap \Theta|_d}{|C_{\ell,\boldsymbol{k}}|_d}\right) \bar{\mathbf{f}}_{\ell,\boldsymbol{k}}(t+\Delta t),$$

following the direction of [Mohamad and Succi, 2009], called "equilibrium scheme" [Krüger et al., 2017, (5.34) Chapter 5]. For the external boundaries, we impose a bounce back boundary condition with the given velocity $(u^\circ, 0)^{\mathsf{t}}$ on the left, top and bottom boundary and a oth order extrapolation boundary condition on the outlet. We take $\lambda = 1$, $\rho_0 = 1$ and all the parameters are set to obtain a Reynolds number $\mathrm{Re} = 1200$.

2.8.2.2.2 Results Due to the important role of the viscosity in the whole domain and not only where the mesh is more refined, we cannot compare the solution at each time frame because the *von Kármán* instability is going to develop at different times because its triggering is essentially of numerical origin. Thus, we analyze some integral quantities [Eitel-Amor et al., 2013, Fakhari and Lee, 2015] like the drag coefficient C_D, the lift coefficient C_L and the Strouhal number St, given by:

$$C_D = \frac{2F_1}{\rho_0 (u^\circ)^2 \sqrt{\int_\Theta \mathrm{d}\boldsymbol{x}}}, \qquad C_L = \frac{2F_2}{\rho_0 (u^\circ)^2 \sqrt{\int_\Theta \mathrm{d}\boldsymbol{x}}}, \qquad \mathrm{St} = \frac{\omega \sqrt{\int_\Theta \mathrm{d}\boldsymbol{x}}}{u^\circ},$$

where $\boldsymbol{F} = (F_1, F_2)^{\mathsf{t}}$ is the total force acting on the obstacle Θ and ω is the shedding frequency of the vortices. The shedding frequency is computed using the fast Fourier transform on the available lift coefficient. We are interested in comparing these dimensionless quantities between the reference method on a uniform mesh and the adaptive method on the evolving adaptive mesh, as well as the occupation rates introduced in Section 2.8.2.1.

We fix $\underline{\ell} = 2$ and we consider the set of maximum level/threshold $(\overline{\ell}, \epsilon) = (7, 7.5\mathrm{e}{-4}), (8, 3.75\mathrm{e}{-4})$ and $(9, 1.75\mathrm{e}{-4})$ and $\overline{\mu} = 1$ for the mesh refinement. The reason why we have chosen to decrease ϵ as $\overline{\ell}$ increases shall be clear in a moment.

In Figure 2.34, we observe an excellent agreement on the value of the integral quantities between the adaptive method and the reference method. The discrepancies are obviously reduced as $\overline{\ell}$ increases because of the variation of the respective threshold parameter ϵ. This comes from the fact that the most important contribution to these quantities comes from the area around the obstacle Θ, where the flow regime can be considered to be, to some extent, highly inertial (or hyperbolic). For this kind of regime, previous works [Cohen et al., 2003] and the results of Section 2.8.1 and Section 2.8.2.1 have shown that the Harten heuristics is respected with our choice of $\mathcal{T}_\epsilon$ and $\mathcal{H}_\epsilon$. The occupation rates are interesting and become better and better with $\overline{\ell}$ as we also observed for the solution of the Euler system in Section 2.8.2.1, despite the fact that we reduced the threshold parameter ϵ as we increased $\overline{\ell}$. The initial growth of these rates followed by a decrease is due to some initial acoustic waves quickly propagating radially in the domain which are eventually damped by the external boundary conditions and are inherent to the lattice Boltzmann method and its way of treating boundary conditions. These waves do not affect computations on a longer time scale. The values of the occupation rates stabilize after the complete onset of the instability followed by the periodic regime. On the opposite side, far from the obstacle and close to the outlet, the regime can be considered to be mostly diffusive (or parabolic) and here the Harten heuristics could be violated. The purpose behind the division of the parameter ϵ by two at each time we increased the maximum level $\overline{\ell}$ was to try to follow

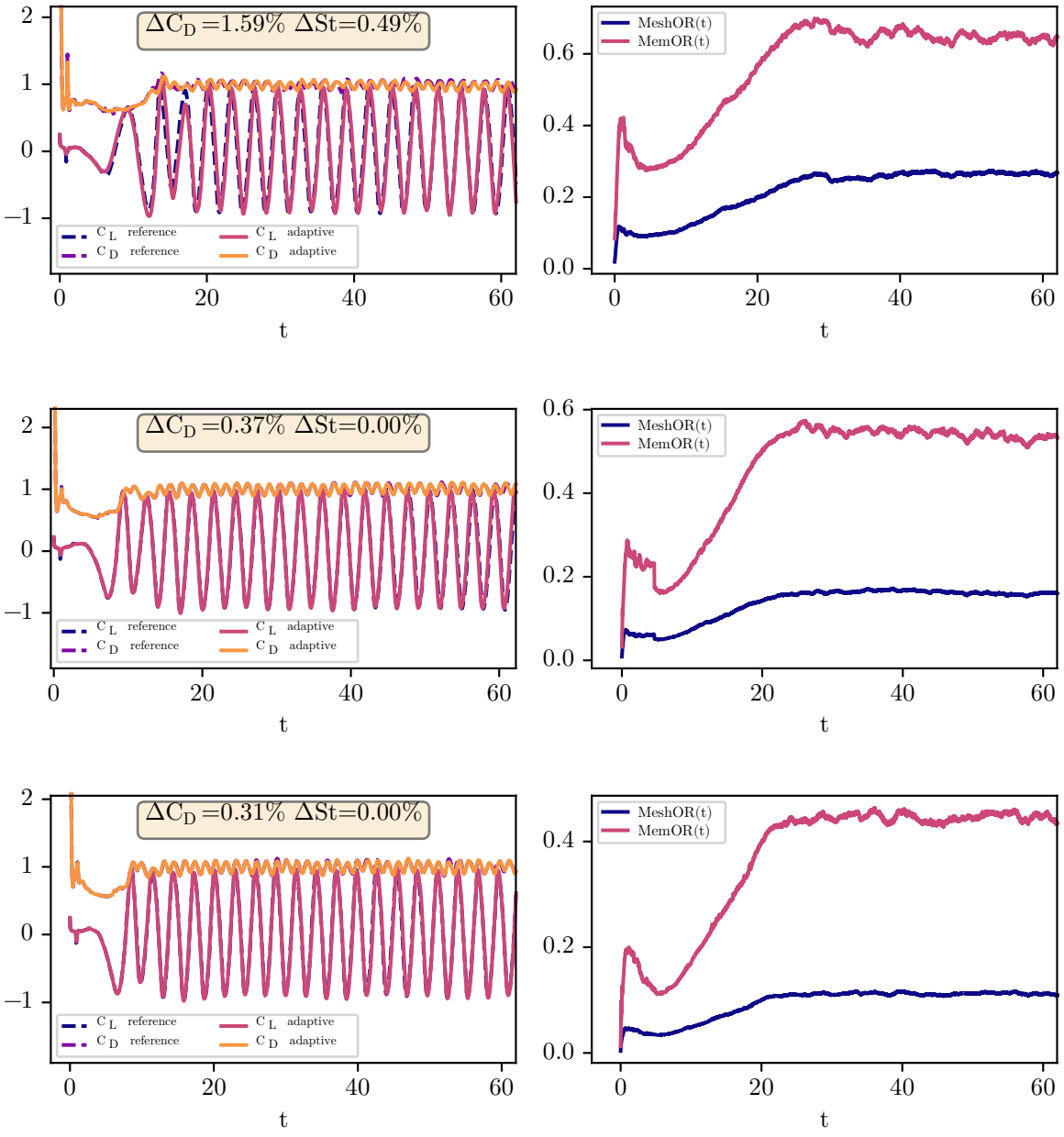


Figure 2.34: On the left, time behavior of the drag and lift coefficient for the reference and the adaptive scheme and on the right, mesh and memory occupation rates, for Re = 1200, $\underline{\ell} = 2$ and $\overline{\mu} = 1$. From top to bottom: $(\overline{\ell}, \epsilon) = (7, 7.5\text{e}-4), (8, 3.75\text{e}-4)$ and $(9, 1.75\text{e}-4)$. The drag coefficient has been normalized with its average for the reference scheme and the lift coefficient with the maximum for the reference scheme.

the vortices until they reach the outlet.

As shown in Figure 2.35, Figure 2.36, Figure 2.37, this attempt of following the vortices until the outlet with the finest resolution $\overline{\ell}$ was only partially successful. More investigations are needed to clarify the role of the oth order extrapolation boundary conditions coupled with the multiresolution procedure. This is not surprising by looking at (2.26) and by concluding that the exact solution of the Navier-Stokes equations for this Reynolds number must have more than only one bounded derivative (*i.e.* more than just bounded vorticity), because the details scale of a factor larger than two at each change of level. This is related to the nature of the solution and we refer to [N'guessan, 2020] for a detailed study of the use of multiresolution in order to solve the incompressible Navier-Stokes equations. It is also worthwhile mentioning that the D_2Q_9 scheme for the Navier-Stokes system under acoustic scaling $\Delta t \propto \Delta x$ is not converging for $\Delta x \to 0$: therefore one should be really careful once comparing the results for different $\overline{\ell}$, as we did.

To conclude, this test shows that our method is effective when applied to parabolic problems solved with the lattice Boltzmann method. This reflects on the excellent results concerning the integral quantities. The investigations of the following Section 3.1 show that our adaptive method does not modify (for $\gamma \geq 1$) the viscosity of the flow, thus confirming once more that it is suitable to simulate the Navier-Stokes equations. Furthermore, the method also preserves higher order terms and thus is less likely to modify the stability properties of the reference scheme.

2.8.2.3 D_3Q_6 for one conservation law

Since realistic problems are 3D, we propose a final demonstration with $d = 3$.

2.8.2.3.1 The problem and the scheme We consider the approximation of the solution of the initial-value problem for the linear transport equation

$$\begin{cases} \partial_t u + \partial_{x_1}(V_1 u) + \partial_{x_2}(V_2 u) + \partial_{x_3}(V_3 u) = 0, & t \in [0,T], \quad \boldsymbol{x} \in \mathbb{R}^3, \\ u(0,\boldsymbol{x}) = \mathbb{1}_{\sqrt{x_1^2+x_2^2+x_3^2} \leq 0.15}(\boldsymbol{x}), & \boldsymbol{x} \in \mathbb{R}^3, \end{cases}$$

where the scalar u is transported with velocity $\boldsymbol{V} = (V_1, V_2, V_3)^{\mathrm{t}}$.

As numerical scheme, we use a D_3Q_6 scheme with $N = 1$—*cf.* [Feldhusen et al., 2016]—corresponding to the choice of discrete velocities $\boldsymbol{c}_1 = (1,0,0)^{\mathrm{t}}$, $\boldsymbol{c}_2 = (-1,0,0)^{\mathrm{t}}$, $\boldsymbol{c}_3 = (0,1,0)^{\mathrm{t}}$, $\boldsymbol{c}_4 = (0,-1,0)^{\mathrm{t}}$, $\boldsymbol{c}_5 = (0,0,1)^{\mathrm{t}}$ and $\boldsymbol{c}_6 = (0,0,-1)^{\mathrm{t}}$, and moment matrix

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ \lambda & -\lambda & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & \lambda & -\lambda \\ \lambda^2 & \lambda^2 & -\lambda^2 & -\lambda^2 & 0 & 0 \\ \lambda^2 & \lambda^2 & 0 & 0 & -\lambda^2 & -\lambda^2 \end{bmatrix},$$

with moments at equilibrium

$$m_2^{\mathrm{eq}}(\mathsf{m}_1) = V_1 \mathsf{m}_1, \qquad m_3^{\mathrm{eq}}(\mathsf{m}_1) = V_2 \mathsf{m}_1, \qquad m_4^{\mathrm{eq}}(\mathsf{m}_1) = V_3 \mathsf{m}_1, \qquad m_5^{\mathrm{eq}}(\mathsf{m}_1) = 0, \qquad m_6^{\mathrm{eq}}(\mathsf{m}_1) = 0.$$

The relaxation parameters are $s_2 = s_3 = s_4 = 1.4$ and $s_5 = s_6 = 1$ with $\lambda = 1$.

2.8.2.4 Results

We carry on the simulation until $T = 0.78125$ with $\underline{\ell} = 1$, $\overline{\ell} = 8$, $\overline{\mu} = 2$ and $\epsilon = 1\mathrm{e}{-3}$. Some snapshots of the solutions and the mesh at different time steps are provided in Figure 2.38: one can appreciate the significantly hollow mesh produced by multiresolution. The time behavior of the occupation rates is presented in Figure 2.39, corresponding to an excellent compression rate of around 97%. This shows that our method is capable of coping with this three dimensional problem properly and to achieve really interesting occupation rates which are crucial to conduct large

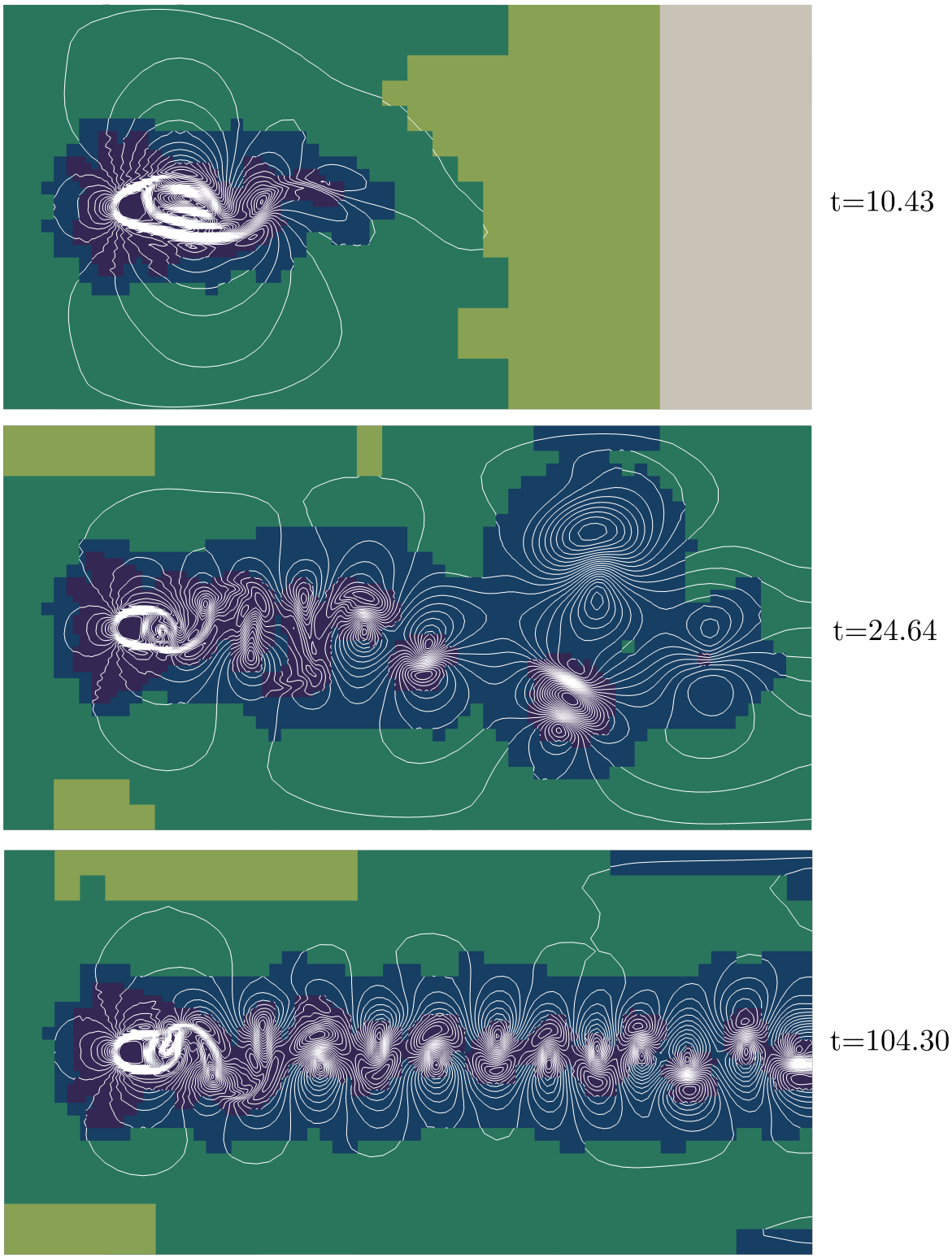


Figure 2.35: Snapshots of the solution of the adaptive scheme for Re = 1200, $\underline{\ell} = 2$, $\overline{\ell} = 7$, $\overline{\mu} = 1$ and $\epsilon = 7.5\text{e}-4$. The colors represent the levels of the mesh and the white contours are that of the velocity modulus.

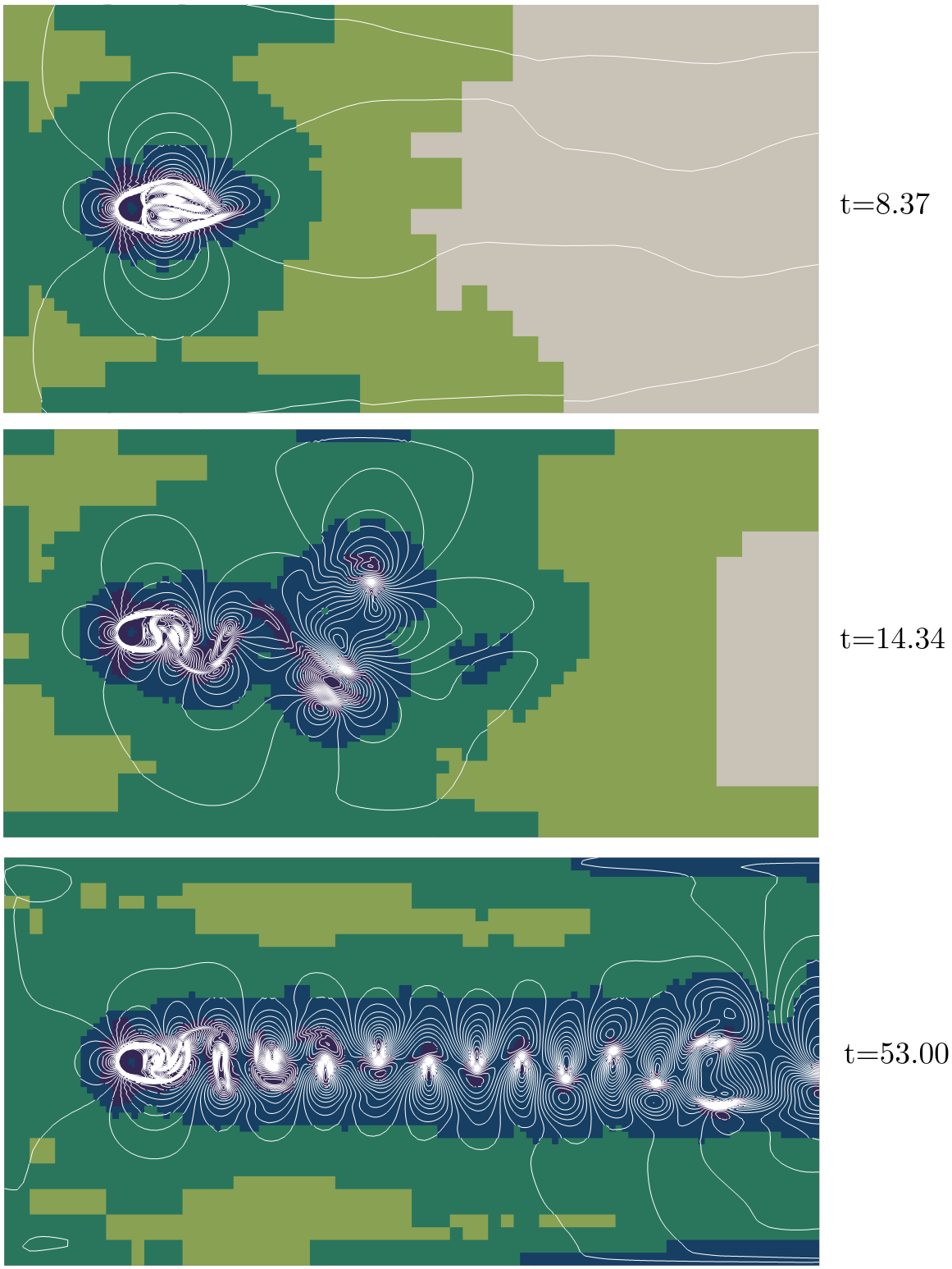


Figure 2.36: Snapshots of the solution of the adaptive scheme for Re = 1200, $\underline{\ell} = 2$, $\overline{\ell} = 8$, $\overline{\mu} = 1$ and $\epsilon = 3.75\text{e}-4$. The colors represent the levels of the mesh and the white contours are that of the velocity modulus.

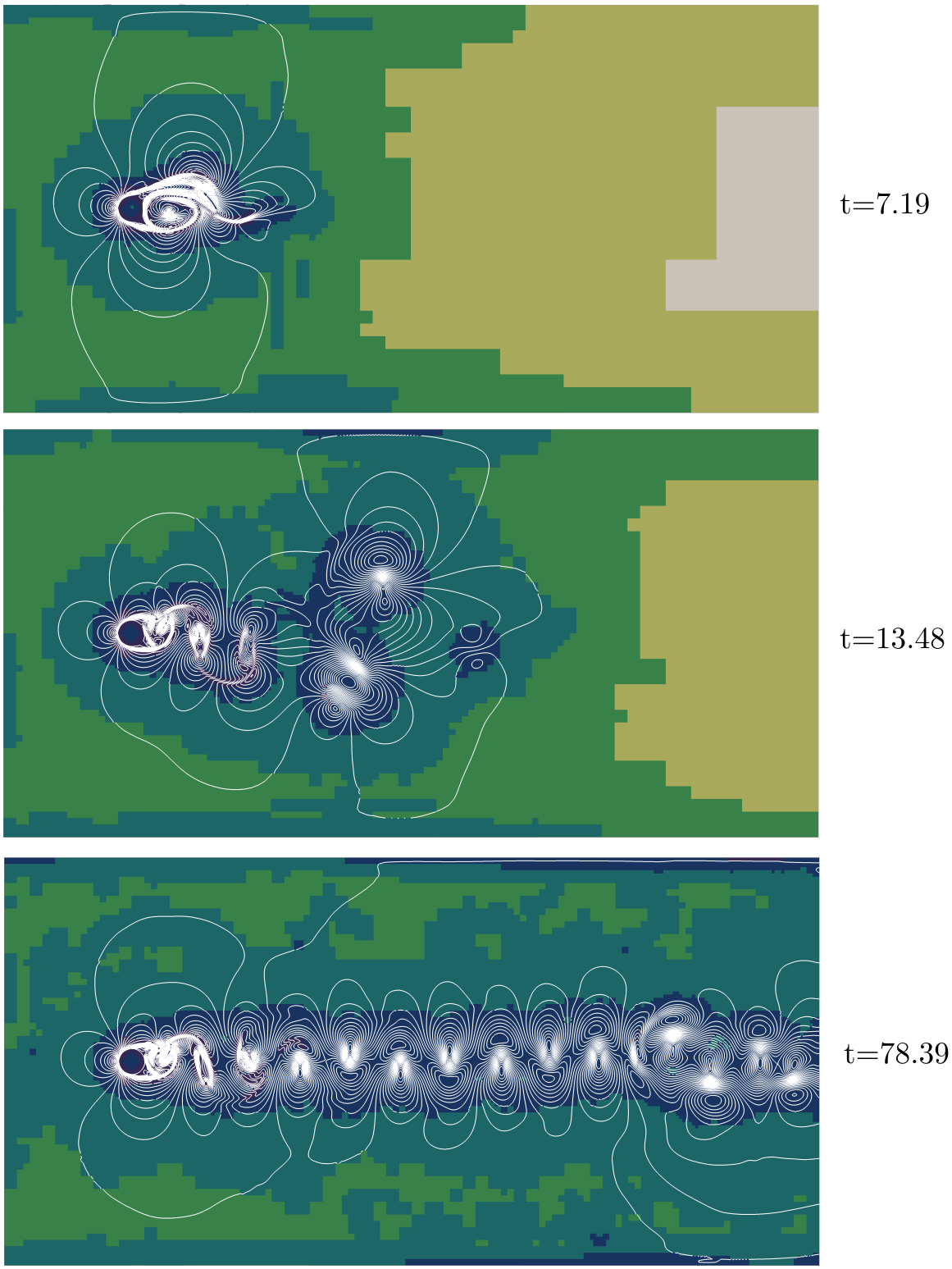


Figure 2.37: Snapshots of the solution of the adaptive scheme for Re = 1200, $\underline{\ell} = 2$, $\overline{\ell} = 9$, $\overline{\mu} = 1$ and $\epsilon = 1.875\text{e}-4$. The colors represent the levels of the mesh and the white contours are that of the velocity modulus.

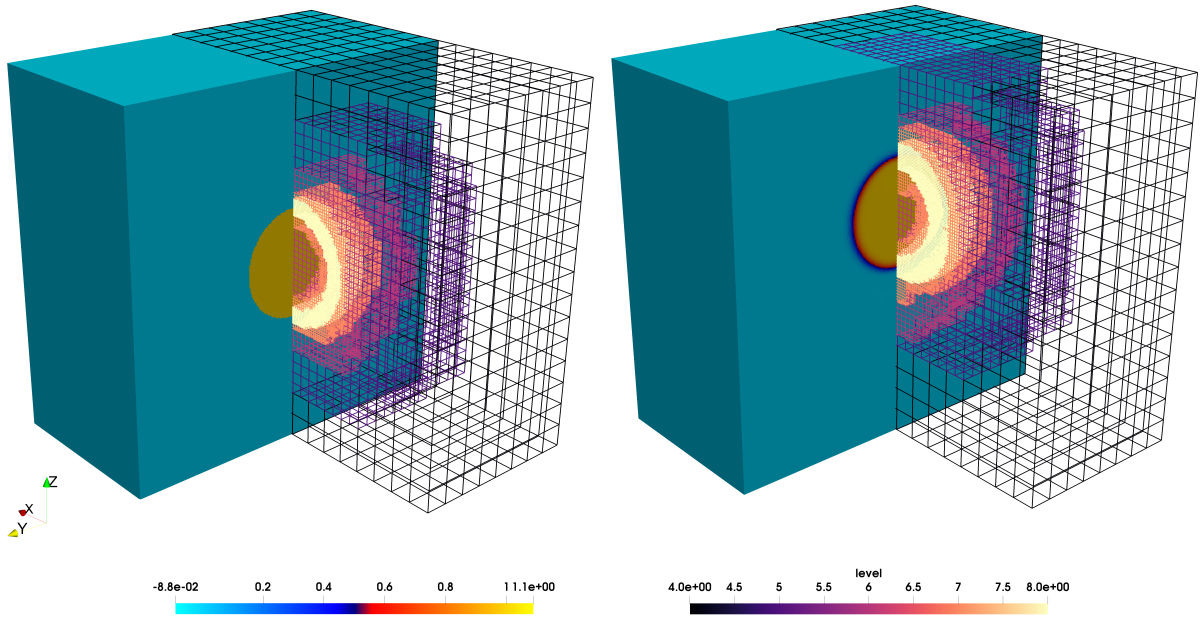


Figure 2.38: Snapshots at times $t = 0$ (left) and $t = 0.3125$ (right) of the solution of the adaptive scheme for the 3D advection equation. The domain has been cut so that the left half of each snapshots shows the value of the conserved variable m_1 whereas the right half represents the structure of the mesh with the corresponding local level of resolution.

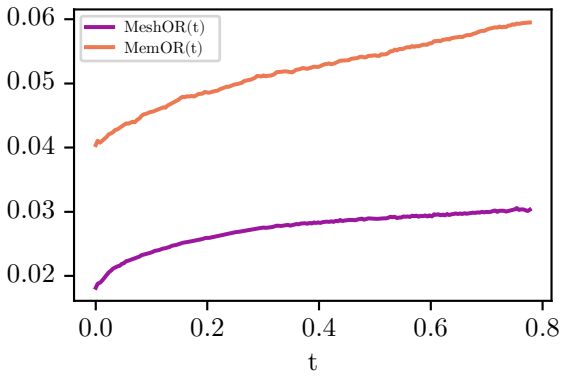


Figure 2.39: Temporal variation of the occupation rates for the adaptive scheme simulating the 3D advection equation.

simulations for 3D problems. Indeed, the full problem would have needed $256^3 \sim 16$ millions cells, which is quite demanding for a sequential code. Thanks to multiresolution, we have made the problem easily treatable with at most $0.03 \times 256^3 \sim 50000$ cells involved (for all sizes). This means that the adaptive mesh occupies 3% of its original size, which is quite impressive and shows the interest of the method to be used in 3D problems, where limitations imposed by the storage of data may become prohibitive.

2.9 Conclusions of Chapter 2

In Chapter 2, we have first proposed a general form of lattice Boltzmann scheme on control volumes making up a uniform mesh. Then, we have performed a mesh coarsening at each time step, using adaptive multiresolution on the solution of the numerical scheme. Before finally adapting the mesh at hand, we have added a refinement step. These two combined steps have ensured to dynamically adapt the grids as the simulations proceed. We have devised lattice Boltzmann schemes based on the original scheme on uniform meshes to work on the mesh that has previously been adapted. We have found—under appropriate assumptions—error bounds on the additional error introduced by mesh adaptation and the adaptive lattice Boltzmann scheme. We have explained how to implement boundary conditions. Finally, we have extensively tested our method on 1D/2D/3D problems.

The error estimates have been correctly retrieved in simulations, both for linear and non-linear problems regardless of the space dimension. Moreover, the obtained memory compression rates are very good for problems with shocks. The abundant tests have shown that the strategy works for very different lattice Boltzmann schemes irrespective of the problem to solve. Moreover, tests have demonstrated that the so-called "leaves collision" yields results with very good accuracy in most of the cases. Alternative collision strategies have been investigated, yielding only marginally better performances in terms of error.

One point that is left for future investigations is the reshaping of the method to be used on complex geometries. More importantly, the peculiar way of enforcing the physics on lattice Boltzmann schemes (*e.g.* the diffusion for the D_2Q_9 under acoustic scaling) calls for a more precise description of the perturbations introduced by the adaptive method. This will be the topic of Section 3.1.

Chapter 3

Further analyses of the adaptive lattice Boltzmann methods: from quantification of the perturbation order through equivalent equations to controlling reflected waves

General context and motivation

In Chapter 2, we have introduced an adaptive lattice Boltzmann method relying on multiresolution and described its behavior concentrating on the role of the threshold parameter ϵ, which allows controlling the perturbation error introduced by non-uniform meshes. Despite this control, much is left to be quantified concerning the perturbations on the target equations induced by this new lattice Boltzmann strategy. The question of determining the equations approximated by the lattice Boltzmann method has to be discussed with particular care, especially when one wants to introduce diffusion terms with a specific structure. Another important issue within the lattice Boltzmann community—which is interesting to analyze for our adaptive lattice Boltzmann method—concerns spurious reflected waves between cells when using adaptive meshes. This is particularly important in applications such as aeroacoustics. It is known that a wave passing through a grid transition normally splits into two parts: the first one propagating through the interface and the second one which is reflected back. The phenomenological reason for this is the different acoustic impedance of two media made up of grids at different levels of refinement.

State of the art

As far as the analysis of the perturbations on the target equations introduced by the adaptive method is concerned, we are going to rely on the equivalent equations [Dubois, 2008], offering a formal procedure to analyze the consistency of lattice Boltzmann methods relying on Taylor expansions of the stream phase. Many strategies, both for the representation of an adaptive grid and for the construction of adaptive lattice Boltzmann methods have been explored to alleviate the issue of spurious reflected waves, see [Horstmann, 2018] and [Lagrava, 2012] for a review. There is no widespread consensus on a standard test case to analyze the issue of reflected waves: we therefore consider a basic one-dimensional configuration and a two-dimensional acoustic pulse test case from [Astoul et al., 2021, Gendre et al., 2017].

Aims and structure of Chapter 3

The aim of Chapter 3 is to clarify several numerical properties of the previously introduced adaptive lattice Boltzmann methods based on multiresolution. In particular:

- Quantify the perturbation introduced by multiresolution on the behavior of the numerical scheme, knowing that these perturbations could change the physical model approximated by the lattice Boltzmann scheme

and one would like the adaptive scheme to behave as close as possible to the reference scheme it stems from. This is the topic of Section 3.1.

- Quantify the amplitude of the reflected spurious waves at the interface between different mesh resolutions, knowing that this is a common problem when deploying lattice Boltzmann schemes on non-uniform grids. This topic is treated in Section 3.2.

Contents

3.1 Quantification of the perturbation error for the multiresolution lattice Boltzmann method

A first topic to discuss is to characterise the order in the discretization parameter Δx until which our adaptive method does not perturb the reference scheme. Eventually, the aim is to dispose of adaptive schemes behaving as close as possible to the reference scheme. More precisely, the prediction operator $\boldsymbol{P}_{\triangle}$, must be accurate enough so that the adaptive method does not significantly alter the behavior of the numerical scheme performed on the finest level of grid. Otherwise, large deviations from the behavior of the reference scheme are theoretically expected and numerically observed. To this end, in Section 3.1.1, we develop an equivalent equation analysis for the multiresolution lattice Boltzmann method. Here, we also recall the approach to the stream phase by [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016] for comparison purpose. Based on this, comparisons with the reference scheme are conducted in Section 3.1.2 for $d = 1$ and extended to $d = 2$ in Section 3.1.3. The theoretical analysis is confirmed by numerical simulations in Section 3.1.4. Conclusions are drawn in Section 3.1.5.

3.1.1 Equivalent equations for the multiresolution lattice Boltzmann method in 1D

We introduce an equivalent equations [Dubois, 2008, Dubois, 2022] analysis of our method for $d = 1$, knowing that this approach allows—on a fixed mesh—to find the actual behavior of the numerical scheme. This analysis pertains to the way of performing the stream phase and does not take the different models for the collision phase into account. Having left the collision phase untouched (2.38) and the time step being global across levels, this phase does not have an impact on the equivalent equations of the scheme. This is rigorously justified as long as the equilibria are linear functions of the conserved moments.

3.1.1.1 Locally uniform meshes

We decide to conduct the analysis on uniform grids at some level of refinement $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$. The aim of considering locally uniform meshes is to employ Taylor expansions to describe the behavior of the numerical schemes. The analysis is still pertinent for adaptive meshes for the following reasons:

- Given a leaf $C_{\ell,\boldsymbol{k}}$ such that $(\ell, \boldsymbol{k}) \in S(\Lambda(t + \Delta t))$, it is surrounded by enough cells (both leaves or halo cells) at the same level ℓ of refinement. Therefore, the mesh can be considered to be locally uniform [Cohen et al.,

2003, Section 3.5.1 and Remark 3.4], which perfectly fits the local character of the analysis that we want to develop.

- The theoretical analysis that we want to develop relies upon the assumption that the distributions are smooth functions on the whole domain at any considered time. Thus, once one fixes a small but finite tolerance ϵ and the number of authorized level $\Delta\underline{\ell} = \overline{\ell} - \underline{\ell} \geq 0$, letting $\overline{\ell}$ increase (imagine $\overline{\ell} \to +\infty$, hence $\Delta x \to 0$), at some point, even the uniform mesh at level $\underline{\ell}$ will allow to control errors by ϵ, thanks to multiresolution.

3.1.1.2 Lax-Wendroff stream phase

For the purpose of comparing to previous strategies in the literature, we also consider the stream scheme described by [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016]. In our multiresolution framework, constructing the adaptive stream phase (2.40) by using the reconstruction operator was dictated by the wish of recovering a control on the perturbation error of the adaptive method. However, for methods based on the heuristic AMR, difference approaches are indeed possible. The authors of [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016] introduce a Lax-Wendroff stream phase for the D_2Q_9 scheme, while the collision phase remains unchanged, *cf.* (2.38). Consider the discrete velocity indexed by $j \in [\![1, q]\!]$ and let $(\ell, \boldsymbol{k}) \in S(\Lambda(t+\Delta t))$, then the stream phase reads

$$\overline{\mathrm{f}}^{j}_{\ell,\boldsymbol{k}}(t+\Delta t) = \left(1 - \frac{1}{4^{\Delta\ell}}\right)\overline{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}}(t) + \frac{1}{2^{\Delta\ell+1}}\left(1 + \frac{1}{2^{\Delta\ell}}\right)\overline{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}-\boldsymbol{c}_j}(t) - \frac{1}{2^{\Delta\ell+1}}\left(1 - \frac{1}{2^{\Delta\ell}}\right)\overline{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}+\boldsymbol{c}_j}(t), \tag{3.1}$$

working for schemes where $\max_{j\in[\![1,q]\!]} \|\boldsymbol{c}_j\|_\infty \leq 1$, such as the D_2Q_4 scheme of Section 1.5.3 and the D_2Q_9 scheme of Section 1.5.4. For $d = 1$ and any scheme with $\max_{j\in[\![1,q]\!]} |c_j| \leq 2$, the Lax-Wendroff scheme reads

$$\overline{\mathrm{f}}^{j}_{\ell,k}(t+\Delta t) = \left(1 - \frac{|c_j|^2}{4^{\Delta\ell}}\right)\overline{\mathrm{f}}^{j,\star}_{\ell,k}(t) + \frac{|c_j|}{2^{\Delta\ell+1}}\left(1 + \frac{|c_j|}{2^{\Delta\ell}}\right)\overline{\mathrm{f}}^{j,\star}_{\ell,k-\mathrm{sign}(c_j)}(t) - \frac{|c_j|}{2^{\Delta\ell+1}}\left(1 - \frac{|c_j|}{2^{\Delta\ell}}\right)\overline{\mathrm{f}}^{j,\star}_{\ell,k+\mathrm{sign}(c_j)}(t). \tag{3.2}$$

It can be seen that (3.2) is linked to the prediction operator $\boldsymbol{P}_\triangle$ and the local reconstruction polynomial (2.19) when $\gamma = 1$ (2.21) but is not a multiresolution scheme, because it is not built upon the reconstruction operator resulting from the cascade of prediction operators. This is the meaning of the following result.

Proposition 3.1.1: Origin of (3.2)

Let $d = 1$, then the Lax-Wendroff stream (3.2) is obtained by approximating the terms in the pseudo-fluxes from (2.40) with

$$\mathring{\overline{\mathrm{f}}}^{j,\star}_{\overline{\ell},\overline{k}}(t) \approx \frac{1}{\Delta x}\int_{C_{\overline{\ell},\overline{k}}} \pi^{j,\star}_{\ell,k}(t,x)\mathrm{d}x, \qquad \overline{k} \in \mathcal{E}^{j}_{\ell,k} \cup \mathcal{A}^{j}_{\ell,k},$$

where $\pi^{j,\star}_{\ell,k}(t,\cdot)$ is the local reconstruction polynomial for $\overline{\mathrm{f}}^{j,\star}(t)$ for the cell $C_{\ell,k}$ taking $\gamma = 1$, that is, given by (2.21).

Proof. We do not list the time variable for the sake of notation. Consider $\overline{k} \in \mathcal{E}^{j}_{\ell,k} \cup \mathcal{A}^{j}_{\ell,k}$, then, using (2.21) and a change of variable

$$\begin{aligned}\frac{1}{\Delta x}\int_{C_{\overline{\ell},\overline{k}}} \pi^{j,\star}_{\ell,k}(x)\mathrm{d}x = 2^{\Delta\ell}\Bigg[&\left(-\frac{1}{24}\overline{\mathrm{f}}^{j,\star}_{\ell,k-1} + \frac{13}{12}\overline{\mathrm{f}}^{j,\star}_{\ell,k} - \frac{1}{24}\overline{\mathrm{f}}^{j,\star}_{\ell,k+1}\right)x + \left(-\frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k-1} + \frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k+1}\right)\frac{x^2}{2} \\ &+\left(\frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k-1} - \overline{\mathrm{f}}^{j,\star}_{\ell,k} + \frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k+1}\right)\frac{x^3}{3}\Bigg]^{x=2^\ell(x_{\overline{\ell},\overline{k}}-x_{\ell,k})+1/2^{\Delta\ell+1}}_{x=2^\ell(x_{\overline{\ell},\overline{k}}-x_{\ell,k})-1/2^{\Delta\ell+1}}.\end{aligned}$$

This gives

$$\begin{aligned}\frac{1}{\Delta x}\int_{C_{\overline{\ell},\overline{k}}} \pi^{j,\star}_{\ell,k}(x)\mathrm{d}x = &\left(-\frac{1}{24}\overline{\mathrm{f}}^{j,\star}_{\ell,k-1} + \frac{13}{12}\overline{\mathrm{f}}^{j,\star}_{\ell,k} - \frac{1}{24}\overline{\mathrm{f}}^{j,\star}_{\ell,k+1}\right) + 2^\ell\left(-\frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k-1} + \frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k+1}\right)(x_{\overline{\ell},\overline{k}} - x_{\ell,k}) \\ &+\left(\frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k-1} - \overline{\mathrm{f}}^{j,\star}_{\ell,k} + \frac{1}{2}\overline{\mathrm{f}}^{j,\star}_{\ell,k+1}\right)\left(2^{2\ell}(x_{\overline{\ell},\overline{k}} - x_{\ell,k})^2 + \frac{1}{12\times 2^{2\Delta\ell}}\right).\end{aligned}$$

We consider different velocities. Negative velocities are studied by symmetry. Recall that (2.42) holds, hence

$$\mathcal{E}^j_{\ell,k} = \{k2^{\Delta\ell} - \delta \,:\, \delta \in [\![1, c_j]\!]\}, \qquad \mathcal{A}^j_{\ell,k} = \{(k+1)2^{\Delta\ell} - \delta \,:\, \delta \in [\![1, c_j]\!]\}, \tag{3.3}$$

and we have to consider that

$$\overline{k} = k2^{\Delta\ell} - 1, \quad x_{\overline{\ell},\overline{k}} - x_{\ell,k} = -(2^{-\ell} + 2^{-\overline{\ell}})/2, \tag{3.4}$$

$$\overline{k} = k2^{\Delta\ell} - 2, \quad x_{\overline{\ell},\overline{k}} - x_{\ell,k} = -(2^{-\ell} + 3\times 2^{-\overline{\ell}})/2, \tag{3.5}$$

$$\overline{k} = (k+1)2^{\Delta\ell} - 1, \qquad x_{\overline{\ell},\overline{k}} - x_{\ell,k} = (2^{-\ell} - 2^{-\overline{\ell}})/2, \tag{3.6}$$

$$\overline{k} = (k+1)2^{\Delta\ell} - 2, \qquad x_{\overline{\ell},\overline{k}} - x_{\ell,k} = (2^{-\ell} - 3\times 2^{-\overline{\ell}})/2. \tag{3.7}$$

- Let $c_j = 1$, hence the flux term in (2.40) is approximated by

$$\sum_{\overline{\boldsymbol{k}}\in\mathcal{E}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}} - \sum_{\overline{\boldsymbol{k}}\in\mathcal{A}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}} \approx -\frac{1}{2^{\Delta\ell}}\bar{\mathsf{f}}^{j,\star}_{\ell,k} + \frac{1}{2}\Big(1 + \frac{1}{2^{\Delta\ell}}\Big)\bar{\mathsf{f}}^{j,\star}_{\ell,k-1} - \frac{1}{2}\Big(1 - \frac{1}{2^{\Delta\ell}}\Big)\bar{\mathsf{f}}^{j,\star}_{\ell,k+1},$$

  using (3.4) and (3.6) into (3.3), yielding the claim.
- Let $c_j = 2$, then we obtain

$$\sum_{\overline{\boldsymbol{k}}\in\mathcal{E}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}} - \sum_{\overline{\boldsymbol{k}}\in\mathcal{A}^j_{\ell,\boldsymbol{k}}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{\boldsymbol{k}}} \approx -\frac{4}{2^{\Delta\ell}}\bar{\mathsf{f}}^{j,\star}_{\ell,k} + \Big(1 + \frac{2}{2^{\Delta\ell}}\Big)\bar{\mathsf{f}}^{j,\star}_{\ell,k-1} - \Big(1 - \frac{2}{2^{\Delta\ell}}\Big)\bar{\mathsf{f}}^{j,\star}_{\ell,k+1},$$

  using (3.4), (3.5), (3.6) and (3.7) into (3.3), proving the claim.

□

One might ask whether it could be possible to see (3.2) as a multiresolution scheme by using prediction operators $\boldsymbol{P}_\triangle$ different from (2.16) with $\gamma = 1$. Concerning $\gamma = 0$, (3.2) cannot be obtained from multiresolution because the last term in the scheme involves the neighbor downwind in the direction of the discrete velocity, whereas this prediction operator does not involve any neighboring cell, since $\gamma = 0$. We could envision to use prediction operators based on two values. These are employed, for example, in the point-wise multiresolution [Harten, 1993, Chiavassa and Donat, 2001, Forster, 2016, Soni et al., 2017] but are not suitable to be used with volumetric representations. Considering a local reconstruction polynomial of degree one around $C_{\ell,k}$

$$\pi_{\ell,k}(x) = \pi^0_{\ell,k} + \pi^1_{\ell,k}\Big(\frac{x - x_{\ell,k}}{\Delta x_\ell}\Big)$$

and enforcing that its average yield the exact averages on $C_{\ell,k\pm1}$, we obtain the prediction

$$\hat{\bar{\mathsf{f}}}_{\ell+1,2k+\delta} = \frac{1}{2}\Big(1 + \frac{(-1)^\delta}{4}\Big)\bar{\mathsf{f}}_{\ell,k-1} + \frac{1}{2}\Big(1 - \frac{(-1)^\delta}{4}\Big)\bar{\mathsf{f}}_{\ell,k+1}, \qquad \delta \in \Sigma.$$

The first issue with this operator is that it does not fulfill the consistency property in Definition 2.3.2: the average of the predicted values on the siblings does not give back the value on the parent cell. Thus, it does not yield a conservative multiresolution. Moreover, it cannot yield (3.2) when applied recursively, because otherwise this formula would make use both of the neighbors $k\pm1$ and $k\pm2$. If we look for non-symmetric prediction operators, they degenerate into the case $\gamma = 0$.

This discussion emphasizes the fact that, for the Lax-Wendroff scheme (3.2), the reconstruction polynomial for $\gamma = 1$ is not used in a recursive manner to yield information at the finest level $\overline{\ell}$, thus it does not generate a multiresolution scheme. We shall study this in terms of impact on the quality of the approximation.

3.1.1.3 Reconstruction flattening

In Section 2.5.4.1, we have observed that we can explicitly write the reconstruction in a non-recursive fashion, so that all the computation can be written on the local grid level. Even if, in Section 2.5.4.1, this was essentially a way of

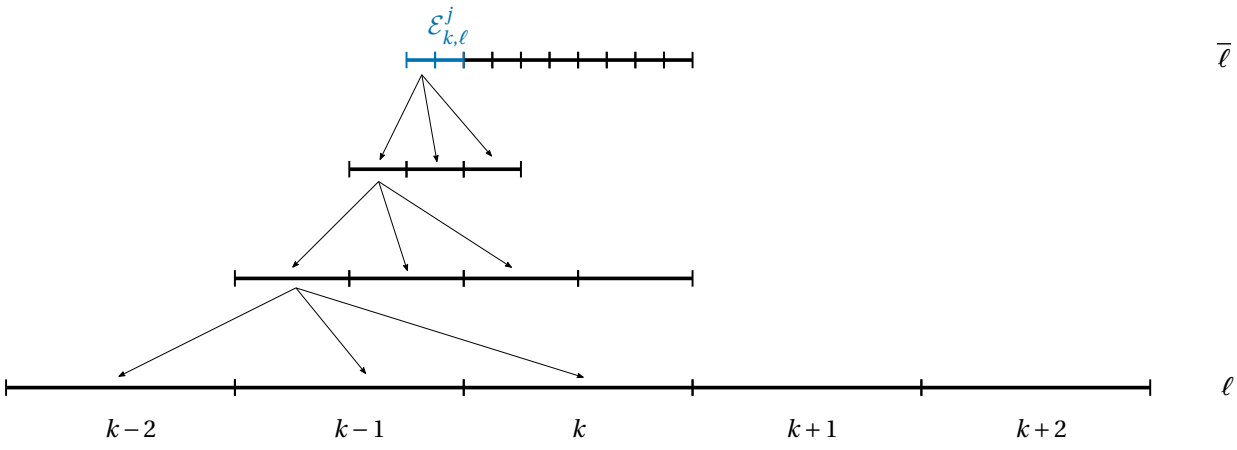


Figure 3.1: Example of flattening procedure for $d = 1$ with $\gamma = 1$ for a velocity $c_j = 2$. The cells in blue correspond to those belonging to $\mathcal{E}^j_{\ell,k}$. The prediction operator is recursively applied (arrows) until reaching the level we are looking for, namely ℓ.

speeding up computations, we now observe that it allows to analyze the adaptive scheme as if we were on a uniform mesh, fostering the possibility of employing Taylor expansions. In what follows, we assume that the discrete velocities of the considered method imply at most two neighbors in each direction, this is $\max_{j\in[\![1,q]\!]} |c_j| \leq 2$. Thus, the analysis covers all the schemes that have been treated in the present work. However, the study can be straightforwardly extended to larger stencils upon considering sums in the formula (3.8) to come spanning larger sets of integers, in particualar $|\delta| \in [\![0, 2\left\lceil \frac{\max_{j\in[\![1,q]\!]} |c_j|}{2} \right\rceil]\!]$. The generalization to $\gamma \geq 2$ is achieved in the same manner, *cf.* Chapter 4.

Flattening the reconstruction operator means that—in the spirit of Section 2.5.4.1 and in particular (2.45)—we are able to compute the set of weights $(F^j_{\Delta\ell,\delta})_{\delta\in[\![-2,2]\!]} \subset \mathbb{R}$ such that (2.40) becomes

$$\overline{\mathsf{f}}^j_{\ell,k}(t+\Delta t) = \overline{\mathsf{f}}^{j,\star}_{\ell,k}(t) + \frac{1}{2^{\Delta\ell}}\left(\sum_{\overline{k}\in\mathcal{E}^j_{\ell,k}} \hat{\hat{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{k}}(t) - \sum_{\overline{k}\in\mathcal{A}^j_{\ell,k}} \hat{\hat{\mathsf{f}}}^{j,\star}_{\overline{\ell},\overline{k}}(t)\right) = \overline{\mathsf{f}}^{j,\star}_{\ell,k}(t) + \frac{1}{2^{\Delta\ell}} \sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta} \overline{\mathsf{f}}^{j,\star}_{\ell,k+\delta}(t), \tag{3.8}$$

The formula (3.8) correspond to what is illustrated in Figure 3.1: we have condensed the computation of the total pseudo-flux at the finest level $\overline{\ell}$ as a weighted sum of values on five neighbors at the current level ℓ.

It is important to observe that both the stream phase of the multiresolution scheme (2.40) and the Lax-Wendroff scheme (3.2) can be put under the formalism introduced in (3.8). The fundamental advantage of this representation is that we can develop (3.8) in Taylor series around the considered cell.

3.1.1.4 Taylor expansion for adaptive schemes

We do this by adopting a Finite Difference point of view [LeVeque, 2002, Chapter 8], thus (3.8) becomes

$$f^j(t+\Delta t, x_{\ell,k}) = f^{j,\star}(t, x_{\ell,k}) + \frac{1}{2^{\Delta\ell}} \sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta} f^{j,\star}(t, x_{\ell,k+\delta}),$$

where we consider that the discrete solution f stems from the point-wise evaluation, at the cell centers $x_{\ell,k}$, of an underlying smooth function f. Expanding in time and space around $(t, x_{\ell,k})$, which is understood, we obtain

$$\begin{aligned}
\sum_{h=0}^{+\infty} \frac{\Delta t^h}{h!} \partial_t^h f^j &= f^{j,\star} + \sum_{h=0}^{+\infty} \frac{2^{\Delta\ell(h-1)}\Delta x^h}{h!} \left(\sum_{\delta=-2}^{+2} \delta^h F^j_{\Delta\ell,\delta}\right) \partial_x^h f^{j,\star} \\
&= \left(1 + \frac{1}{2^{\Delta\ell}} \sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta}\right) f^{j,\star} + \Delta x \left(\sum_{\delta=-2}^{+2} \delta F^j_{\Delta\ell,\delta}\right) \partial_x f^{j,\star} + \Delta x^2 \left(2^{\Delta\ell-1} \sum_{\delta=-2}^{+2} \delta^2 F^j_{\Delta\ell,\delta}\right) \partial_{xx} f^{j,\star} \\
&\quad + \Delta x^3 \left(\frac{2^{2\Delta\ell-1}}{3} \sum_{\delta=-2}^{+2} \delta F^j_{\Delta\ell,\delta}\right) \partial_x^3 f^{j,\star} + O(\Delta x^4).
\end{aligned} \tag{3.9}$$

Observe that the first term in (3.9) concerns the time step Δt. Still, since it is global across levels of resolution, this part of the expansion does not depend on $\Delta\ell$. The aim is to compare (3.9) to an analogous one for the reference scheme (that is, the scheme for $\ell = \overline{\ell}$), in order to study to which extent the adaptive scheme behaves—uniformly in $\Delta\ell$—as the reference scheme. This concerns both the approximated equations and hopefully, the stability conditions (which have to take the specific collision phase into account). Observe however that the equivalent/modified equation have been used—under restrictive assumptions on the space stencil and limited to one-step schemes—to asses the stability of Finite Difference methods [Warming and Hyett, 1974].

3.1.2 Maximal match order between adaptive and reference schemes

We determine until which order—as function of γ (for $\gamma = 0, 1$) or when using the Lax-Wendroff stream (3.2)—the equivalent equations agree with those for the reference method.

3.1.2.1 Target Taylor expansion

To establish this comparison, the target Taylor expansion of the reference scheme (2.14) is obtained—as for the adaptive scheme—considering that the discrete solution stems from the point-wise evaluation of a smooth function. This provides

$$f^j(t+\Delta t, x^-_{\overline{\ell},k}) = f^{j,\star}(t, x^-_{\overline{\ell},k-c_j}) = f^{j,\star}(t, x^-_{\overline{\ell},k} - c_j\Delta x),$$

hence the Taylor expansion gives

$$\sum_{h=0}^{+\infty} \frac{\Delta t^h}{h!}\partial_t^h f^j = \sum_{h=0}^{+\infty} \frac{-(c_j\Delta x)^h}{h!}\partial_x^h f^{j,\star} = f^{j,\star} - c_j\Delta x\partial_x f^{j,\star} + \frac{c_j^2\Delta x^2}{2}\partial_{xx} f^{j,\star} - \frac{c_j^3\Delta x^3}{6}\partial_x^3 f^{j,\star} + O(\Delta x^4). \tag{3.10}$$

This expansion (3.10)—once truncated—is the basic building block of the equivalent equations [Dubois, 2008, Dubois, 2022], which are used to study the consistency and the numerical behavior of the reference lattice Boltzmann scheme. Each power of Δx has a different role in determining the behavior of the scheme, in particular:

- The $O(\Delta x)$ term is what we might call an "inertial" term, because it yields inertial contributions in the approximated model at leading order.
- The $O(\Delta x^2)$ term might be called "diffusive" terms since it result in dissipative contribution in the approximated model, which appear to be proportional to Δx when an acoustic scaling is employed, making up numerical diffusion.
- The $O(\Delta x^3)$ term might be called "dispersive" in analogy with Finite Difference methods. Its physical meaning is less clear than for the other terms but it can have a non-negligible impact on the stability of the lattice Boltzmann method.

In our analysis, we shall only be interested in the right hand side of (3.10) and (3.9) because their left hand sides coincide at any order. This comes from the fact that our algorithm is based on a unique global time step imposed by the finest level of resolution $\overline{\ell}$. This holds whatever the scaling between space and time. Notice that, for a given discrete velocity, the Courant number at finest level $\overline{\ell}$ is $\|\boldsymbol{c}_j\|_1$, whereas at a given level ℓ, it is provided by $\|\boldsymbol{c}_j\|_1/2^{\Delta\ell}$. Therefore, roughly speaking, after $2^{\Delta\ell}$ time steps, information shall have traveled the same distance that it would have traveled at finest level $\overline{\ell}$, namely the size of the current cell times $\|\boldsymbol{c}_j\|_1$, that is $2^{\Delta\ell}\|\boldsymbol{c}_j\|_1\Delta x = \|\boldsymbol{c}_j\|_1\Delta x_\ell$.

3.1.2.2 Maximal match orders

Comparing orders between the expansion (3.9) and the reference expansion (3.10), we introduce the definition of "match" between numerical schemes. Since the equivalent equations describe the actual behavior of numerical scheme, the higher the number of matched terms between reference and adaptive scheme, the closer they are going to behave one compared to the other.

Definition 3.1.1: Match

Let $d = 1$. We say that the adaptive stream phase (2.40)/(3.8) matches that of the reference scheme (2.14) at order $H \in \mathbb{N}^*$ whenever

$$\sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta} = 0, \qquad \sum_{\delta=-2}^{+2} \delta^h F^j_{\Delta\ell,\delta} = \frac{(-c_j)^h}{2^{\Delta\ell(h-1)}}, \qquad h \in [\![1, H]\!],$$

for every level $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$, thus for every $\Delta\ell \geq 0$ and for every discrete velocity $j \in [\![1, q]\!]$.

Having set Definition 3.1.1, we are able to study this property for the multiresolution scheme (2.40)/(3.8) with $\gamma = 0, 1$ and for the Lax-Wendroff scheme (3.2). We start by analyzing the simplest scheme, namely (2.40)/(3.8) when $\gamma = 0$.

Proposition 3.1.2: Match for (2.14) and $\gamma = 0$

Let $d = 1$, $\gamma = 0$ and $\Delta\ell > 0$. The weights $(F^j_{\Delta\ell,\delta})_{\delta \in [\![-2,2]\!]} \subset \mathbb{R}$ in (3.8) for the multiresolution stream (2.40) are given by

$$F^j_{\Delta\ell,0} = -|c_j|, \qquad F^j_{\Delta\ell,-\mathrm{sign}(c_j)} = |c_j|,$$

and those not listed are equal to zero. Therefore, the adaptive stream phase matches that of the reference scheme according to Definition 3.1.1 up to order $H = 1$, that is

$$\sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta} = 0, \qquad \sum_{\delta=-2}^{+2} \delta F^j_{\Delta\ell,\delta} = -c_j.$$

Proof. For this choice of $\gamma = 0$, each application of the prediction operator $\boldsymbol{P}_\triangle$ acts by looking at the value on the parent of the cell it predicts on. Overall, for a cell at the finest level $\overline{\ell}$, it returns the value on its ancestor at level ℓ. Consider $c_j > 0$ for the sake of presentation. According to this discussion:

- For $c_j = 1$, having (3.3), the only reconstructions to estimate for (3.8) are

$$\hat{\hat{\mathrm{f}}}^{j,\star}_{\overline{\ell},k2^{\Delta\ell}-1}(t) = \bar{\mathrm{f}}^{j,\star}_{\ell,k-1}(t), \qquad \hat{\hat{\mathrm{f}}}^{j,\star}_{\overline{\ell},(k+1)2^{\Delta\ell}-1}(t) = \bar{\mathrm{f}}^{j,\star}_{\ell,k}(t),$$

  hence the claim.

- For $c_j = 2$, we have also to consider

$$\hat{\hat{\mathrm{f}}}^{j,\star}_{\overline{\ell},k2^{\Delta\ell}-2}(t) = \bar{\mathrm{f}}^{j,\star}_{\ell,k-1}(t), \qquad \hat{\hat{\mathrm{f}}}^{j,\star}_{\overline{\ell},(k+1)2^{\Delta\ell}-2}(t) = \bar{\mathrm{f}}^{j,\star}_{\ell,k}(t),$$

  giving the claim as well.

The match up to order $H = 1$ directly follows from the values of $F^j_{\Delta\ell,0}$ and $F^j_{\Delta\ell,-\mathrm{sign}(c_j)}$. □

With the multiresolution scheme using $\gamma = 0$, the diffusive term stemming at order $O(\Delta x^2)$ in (3.9) does not match that of the target expansion in (3.10). This can be easily seen by taking $c_j = 1$. Then $2^{\Delta\ell} \sum_{\delta=-2}^{\delta=+2} \delta^2 F^j_{\Delta\ell,\delta} = 2^{\Delta\ell} \neq 1$ for $\Delta\ell > 0$. This has a major consequence on the applicability of the method based on $\gamma = 0$, because it correctly recovers the "inertial" terms but fails in correctly accounting for the "dissipative" terms, yielding a wrong diffusion structure with respect to the equations targeted by the reference method. The well-known $\mathrm{D_2Q_9}$ scheme for the incompressible Navier-Stokes system used in Section 2.8.2.2 is one possible example of lattice Boltzmann scheme which would be deeply altered by picking $\gamma = 0$.

We go on by considering the case $\gamma = 1$, which has been thoroughly investigated in Chapter 2. We are going to see that the limitations of $\gamma = 0$ can be solved by considering a larger prediction stencils: indeed—for most of the applications—employing $\gamma = 1$ is sufficient.

Proposition 3.1.3: Match for (2.14) and $\gamma = 1$

Let $d = 1$, $\gamma = 1$ and $\Delta\ell > 0$. The weights $(F^j_{\Delta\ell,\delta})_{\delta\in[\![-2,2]\!]} \subset \mathbb{R}$ in (3.8) for the multiresolution stream (2.40) are given by the recurrence relations

$$\begin{bmatrix} F^j_{\Delta\ell,-2} \\ F^j_{\Delta\ell,-1} \\ F^j_{\Delta\ell,0} \\ F^j_{\Delta\ell,1} \\ F^j_{\Delta\ell,2} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -1/8 & 0 & 0 & 0 \\ 2 & 9/8 & 0 & -1/8 & 0 \\ 0 & 9/8 & 2 & 9/8 & 0 \\ 0 & -1/8 & 0 & 9/8 & 2 \\ 0 & 0 & 0 & -1/8 & 0 \end{bmatrix}}_{=:\boldsymbol{P}} \begin{bmatrix} F^j_{\Delta\ell-1,-2} \\ F^j_{\Delta\ell-1,-1} \\ F^j_{\Delta\ell-1,0} \\ F^j_{\Delta\ell-1,1} \\ F^j_{\Delta\ell-1,2} \end{bmatrix},$$

where the initialization is given by $F^j_{0,-c_j} = 1$ and $F^j_{0,0} = -1$ and the remaining terms are equal to zero. Therefore, the adaptive stream phase matches that of the reference scheme according to Definition 3.1.1 up to order $H = 3$, that is

$$\sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta} = 0, \qquad \sum_{\delta=-2}^{+2} \delta F^j_{\Delta\ell,\delta} = -c_j, \qquad \sum_{\delta=-2}^{+2} \delta^2 F^j_{\Delta\ell,\delta} = \frac{c_j^2}{2^{\Delta\ell}}, \qquad \sum_{\delta=-2}^{+2} \delta^3 F^j_{\Delta\ell,\delta} = -\frac{c_j^3}{2^{2\Delta\ell}}.$$

Proof. In this proof, we omit the time t for the sake of readability. The initialization given in the claim trivially yields the reference scheme. Assume to know the weights for the stream at level $\ell+1$, that is, for $\Delta\ell - 1$. The trick is to observe that, under the assumption $\max_{j\in[\![1,q]\!]} |c_j| \le 2$, the total pseudo-flux on the parent cell equals the sum of the total pseudo-fluxes on the children:

$$\sum_{\overline{k}\in\mathcal{E}^j_{\ell,k}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}} - \sum_{\overline{k}\in\mathcal{A}^j_{\ell,k}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}} = \Bigg(\sum_{\overline{k}\in\mathcal{E}^j_{\ell+1,2k}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}} - \sum_{\overline{k}\in\mathcal{A}^j_{\ell+1,2k}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}}\Bigg) + \Bigg(\sum_{\overline{k}\in\mathcal{E}^j_{\ell+1,2k+1}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}} - \sum_{\overline{k}\in\mathcal{A}^j_{\ell+1,2k+1}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}}\Bigg).$$

Using the recurrence assumption, followed by a change of indices in the sums yield

$$\begin{aligned}\sum_{\overline{k}\in\mathcal{E}^j_{\ell,k}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}} - \sum_{\overline{k}\in\mathcal{A}^j_{\ell,k}} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell,\overline{k}} &= \sum_{\delta=-2}^{+2} F^j_{\Delta\ell-1,\delta} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell+1,2k+\delta} + \sum_{\delta=-2}^{+2} F^j_{\Delta\ell-1,\delta} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell+1,2k+1+\delta} \\ &= \sum_{\delta=-2}^{+2} F^j_{\Delta\ell-1,\delta} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell+1,2k+\delta} + \sum_{\delta=-1}^{+3} F^j_{\Delta\ell-1,\delta-1} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell+1,2k+\delta} = \sum_{\delta=-2}^{+3} \tilde{F}^j_{\Delta\ell-1,\delta} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell+1,2k+\delta},\end{aligned}$$

where we need to use the prediction operator since we know that the local level of refinement is ℓ (and not $\ell+1$) and we have set

$$\tilde{F}^j_{\Delta\ell-1,\delta} = \begin{cases} F^j_{\Delta\ell-1,-2}, & \delta = -2, \\ F^j_{\Delta\ell-1,\delta} + F^j_{\Delta\ell-1,\delta-1}, & \delta \in [\![-1,2]\!], \\ F^j_{\Delta\ell-1,2}, & \delta = 3. \end{cases}$$

Using the expression for the prediction operator, namely (2.22), we obtain

$$\begin{aligned}\sum_{\delta=-2}^{+3} \tilde{F}^j_{\Delta\ell-1,\delta} \hat{\bar{\mathsf{f}}}^{j,\star}_{\ell+1,2k+\delta} &= \tilde{F}^j_{\Delta\ell-1,-2}\Big(\bar{\mathsf{f}}^{j,\star}_{\ell,k-1} + \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k-2} - \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k}\Big) + \tilde{F}^j_{\Delta\ell-1,-1}\Big(\bar{\mathsf{f}}^{j,\star}_{\ell,k-1} - \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k-2} + \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k}\Big) \\ &\quad + \tilde{F}^j_{\Delta\ell-1,0}\Big(\bar{\mathsf{f}}^{j,\star}_{\ell,k} + \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k-1} - \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k+1}\Big) + \tilde{F}^j_{\Delta\ell-1,1}\Big(\bar{\mathsf{f}}^{j,\star}_{\ell,k} - \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k-1} + \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k+1}\Big) \\ &\quad + \tilde{F}^j_{\Delta\ell-1,2}\Big(\bar{\mathsf{f}}^{j,\star}_{\ell,k+1} + \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k} - \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k+2}\Big) + \tilde{F}^j_{\Delta\ell-1,3}\Big(\bar{\mathsf{f}}^{j,\star}_{\ell,k+1} - \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k} + \tfrac{1}{8}\bar{\mathsf{f}}^{j,\star}_{\ell,k+2}\Big).\end{aligned}$$

Substituting the original weights $F^j_{\Delta\ell-1,\delta}$ to $\tilde{F}^j_{\Delta\ell-1,\delta}$ provides, after tedious computations, the final form of the

recurrence relation, namely

$$\sum_{\delta=-2}^{+3} \tilde{F}^j_{\Delta\ell-1,\delta} \hat{\bar{\mathrm{f}}}^{j,\star}_{\ell+1,2k+\delta} = \Big(-\tfrac{1}{8}F^j_{\Delta\ell-1,-1}\Big)\bar{\mathrm{f}}^{j,\star}_{\ell,k-2} + \Big(2F^j_{\Delta\ell-1,-2} + \tfrac{9}{8}F^j_{\Delta\ell-1,-1} - \tfrac{1}{8}F^j_{\Delta\ell-1,1}\Big)\bar{\mathrm{f}}^{j,\star}_{\ell,k-1} + \Big(\tfrac{9}{8}F^j_{\Delta\ell-1,-1} + 2F^j_{\Delta\ell-1,0} + \tfrac{9}{8}F^j_{\Delta\ell-1,1}\Big)\bar{\mathrm{f}}^{j,\star}_{\ell,k} + \Big(-\tfrac{1}{8}F^j_{\Delta\ell-1,-1} + \tfrac{9}{8}F^j_{\Delta\ell-1,1} + 2F^j_{\Delta\ell-1,2}\Big)\bar{\mathrm{f}}^{j,\star}_{\ell,k+1} + \Big(-\tfrac{1}{8}F^j_{\Delta\ell-1,1}\Big)\bar{\mathrm{f}}^{j,\star}_{\ell,k+2}.$$

This provides the matrix $\boldsymbol{P}$. The matching conditions are then proved by recurrence on $\Delta\ell$. For $\Delta\ell = 0$, they trivially hold. Assume that they are true for $\Delta\ell - 1$, then

- For $h = 0$, we have $\sum_{\delta=-2}^{\delta=+2} F^j_{\Delta\ell,\delta} = 2\sum_{\delta=-2}^{\delta=+2} F^j_{\Delta\ell-1,\delta} = 0$ by recurrence assumption.
- For $h = 1$, we have $\sum_{\delta=-2}^{\delta=+2} \delta F^j_{\Delta\ell,\delta} = \sum_{\delta=-2}^{\delta=+2} \delta F^j_{\Delta\ell-1,\delta} = -c_j$ by recurrence assumption.
- For $h = 2$, we have $\sum_{\delta=-2}^{\delta=+2} \delta^2 F^j_{\Delta\ell,\delta} = \frac{1}{2}\sum_{\delta=-2}^{\delta=+2} \delta^2 F^j_{\Delta\ell-1,\delta} = \frac{c_j^2}{2^{\Delta\ell}}$ by recurrence assumption.
- For $h = 3$, we have $\sum_{\delta=-2}^{\delta=+2} \delta^3 F^j_{\Delta\ell,\delta} = \frac{1}{4}\sum_{\delta=-2}^{\delta=+2} \delta^3 F^j_{\Delta\ell-1,\delta} = -\frac{c_j^3}{2^{2\Delta\ell}}$ by recurrence assumption,

completing the proof. □

Once more, we cannot go further in matching the target expansion (3.10) because, for example, we obtain—using the form of $\boldsymbol{P}$—the expression $\sum_{\delta=-2}^{\delta=+2} \delta^4 F^j_{\Delta\ell,\delta} = 2F^j_{\Delta\ell-1,-2} - F^j_{\Delta\ell-1,-1} - F^j_{\Delta\ell-1,1} + 2F^j_{\Delta\ell-1,2}$, proving that the match condition at fourth order cannot be satisfied. Proposition 3.1.3 means that the multiresolution method for $\gamma = 1$ can be successfully employed in contexts where we want to control both the "inertial" and the "diffusive" terms in the equivalent equations, like in the $\mathrm{D_2Q_9}$ scheme under acoustic scaling for the quasi-incompressible Navier-Stokes system, *cf.* Section 2.8.2.2. Moreover, since we also match the target expansion according to Definition 3.1.1 at order $h = 3$, the achievement accomplished on the reference scheme at this order are also preserved by the adaptive scheme. This is a highly desirable feature for scientists who have a good understanding of their reference scheme and who would like to employ our adaptive strategy as a black-box. Even if we are performing an asymptotic analysis in the limit of small space-steps $\Delta x \to 0$, thus corresponding to a low-frequency study, we conjecture that this feature yields lower discrepancies in terms of stability constraints, which need to take the whole spectrum of frequencies and the collision phase at hand into consideration. We performed the whole stability analysis (not presented in this dissertation) for the linear $\mathrm{D_1Q_2}$ and the choice $\gamma = 1$ was indeed the one yielding the least important discrepancies in terms of stability constraints compared to the reference scheme.

To conclude the one-dimensional analysis, we analyze the match for the Lax-Wendroff stream phase (3.2).

Proposition 3.1.4: Match for (3.2)

Let $d = 1$ and $\Delta\ell > 0$. The weights $(F^j_{\Delta\ell,\delta})_{\delta\in[\![-2,2]\!]} \subset \mathbb{R}$ in (3.8) for the Lax-Wendroff stream (3.2) are given by

$$F^j_{\Delta\ell,0} = -\frac{|c_j|^2}{2^{\Delta\ell}}, \qquad F^j_{\Delta\ell,-\mathrm{sign}(c_j)} = \frac{|c_j|}{2}\Big(1 + \frac{|c_j|}{2^{\Delta\ell}}\Big), \qquad F^j_{\Delta\ell,\mathrm{sign}(c_j)} = -\frac{|c_j|}{2}\Big(1 - \frac{|c_j|}{2^{\Delta\ell}}\Big)$$

and the remaining ones are equal to zero. Therefore, the adaptive stream phase matches that of the reference scheme according to Definition 3.1.1 up to order $H = 2$, that is

$$\sum_{\delta=-2}^{+2} F^j_{\Delta\ell,\delta} = 0, \qquad \sum_{\delta=-2}^{+2} \delta F^j_{\Delta\ell,\delta} = -c_j, \qquad \sum_{\delta=-2}^{+2} \delta^2 F^j_{\Delta\ell,\delta} = \frac{c_j^2}{2^{\Delta\ell}}.$$

Proof. The expressions for the weights follow immediately from a direct inspection of (3.2). The match conditions are obtained as usual by using the explicit expressions of the weights. □

However as expected from such a kind of scheme, the dispersive order, namely third-order, is not matched, because we obtain $\sum_{\delta=-2}^{\delta=+2} \delta^3 F^j_{\Delta\ell,\delta} = -c_j \neq -c_j^3/2^{2\Delta\ell}$. This is not trivially evident when considering Proposition 3.1.1, because the stream phase (3.2) is built using the same reconstruction polynomial as the multiresolution scheme

(2.40) for $\gamma = 1$. Still, what changes is that the multiresolution approach employs the reconstruction operator with recursive application of the prediction operator $\boldsymbol{P}_\triangle$ until reaching the finest level $\overline{\ell}$, whereas the Lax-Wendroff scheme uses the local reconstruction polynomial at level ℓ only once to obtain fluxes which need to be computed at the finest level $\overline{\ell}$.

3.1.2.3 Conclusions

To summarize the findings of Section 3.1.1, we have seen that in the case of multiresolution scheme, the prediction stencil γ has to be taken large enough in order to match a sufficient number of desired orders according to Definition 3.1.1. In particular, the number of matched orders is equal to $2\gamma+1$, which is of course a consequence of Proposition 2.3.1 concerning the accuracy of the prediction operator. Therefore, for most of the applications, $\gamma = 0$ is not enough, because it modifies the second-order terms which are frequently used to model diffusion phenomena. Quite the opposite, $\gamma = 1$ is often sufficient for most of the applications and its reliability on the third-order terms is an interesting "icing on the cake". Finally, the Lax–Wendroff scheme (3.2), which is not a multiresolution scheme, matches sharply until order two as claimed in [Fakhari and Lee, 2015] without explicit proof, so it successfully handles diffusive terms but can lead to different stability constraints and oscillatory behavior for the scheme, due to its intrinsic dispersive nature. The reader might have noticed that we have done half of the work compared to [Dubois, 2008, Dubois, 2022], because we perform only the Taylor expansion of the stream phase (2.40) but we do not couple it with the collision phase (2.13) to recover the final expression for the equivalent PDEs on the conserved moments. However, at least in the linear framework, since the collision phases (2.37) and (2.38) keep (2.13) untouched and we have fully characterized the perturbation of the original lattice Boltzmann scheme, which only affects the stream phase, the procedure by [Dubois, 2008, Dubois, 2022] can be easily implemented for the time-space scaling at hand starting from (3.9) instead of (3.10). The computations for the matched terms shall not be repeated since they are the same as for the reference scheme.

3.1.3 Extension of equivalent equations in 2D

So far, we have analyzed the adaptive schemes with the help of the equivalent equations once the reconstruction flattening (3.8) is done. In Section 3.1.3, we show how the previous analysis can be extended to the multidimensional case $d \geq 2$—which has been thoroughly analyzed and employed in Chapter 2—by exploiting the tensorial product structure of the prediction operator $\boldsymbol{P}_\triangle$. The conclusions we can draw are the same as for $d = 1$, namely that the (2.40) for $\gamma = 0$ perturbs starting from second-order, (2.40) for $\gamma = 1$ does so from fourth-order and finally the Lax-Wendroff approach (3.2) perturbs from third-order. As in Section 3.1.1, the analysis holds as long as $\max_{j\in[\![1,q]\!]} \|\boldsymbol{c}_j\|_\infty \leq 2$. For the sake of presentation, we present the case $d = 2$.

The reference stream phase (2.14) rewritten in a Finite Difference fashion reads

$$f^j(t+\Delta t, \boldsymbol{x}^-_{\ell,\boldsymbol{k}}) = f^{j,\star}(t, \boldsymbol{x}^-_{\ell,\boldsymbol{k}-\boldsymbol{c}_j}) = f^{j,\star}(t, \boldsymbol{x}^-_{\ell,\boldsymbol{k}} - \boldsymbol{c}_j\Delta x),$$

thus a Taylor expansion, assuming that we are allowed to commute partial derivatives along different axes by virtue of the Schwarz theorem, gives

$$\begin{aligned}
&\sum_{h=0}^{+\infty} \frac{\Delta t^h}{h!}\partial_t^h f^j = \sum_{h_1=0}^{+\infty}\sum_{h_2=0}^{+\infty} \frac{(-\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^{h_1}(-\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^{h_2}\Delta x^{h_1+h_2}}{h_1!h_2!}\partial_{x_1}^{h_1}\partial_{x_2}^{h_2} f^{j,\star} \qquad (3.11)\\
&= f^{j,\star} - \Delta x((\boldsymbol{c}_j\cdot\boldsymbol{e}_1)\partial_{x_1} + (\boldsymbol{c}_j\cdot\boldsymbol{e}_2)\partial_{x_2})f^{j,\star} + \frac{\Delta x^2}{2}((\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^2\partial_{x_1x_1} + (\boldsymbol{c}_j\cdot\boldsymbol{e}_1)(\boldsymbol{c}_j\cdot\boldsymbol{e}_2)\partial_{x_1x_2} + (\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^2\partial_{x_2x_2})f^{j,\star}\\
&- \frac{\Delta x^3}{6}((\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^3\partial^3_{x_1} + 3(\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^2(\boldsymbol{c}_j\cdot\boldsymbol{e}_2)\partial_{x_1x_1x_2} + 3(\boldsymbol{c}_j\cdot\boldsymbol{e}_1)(\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^2\partial_{x_1x_2x_2} + (\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^3\partial^3_{x_2})f^{j,\star} + O(\Delta x^4).
\end{aligned}$$

This expansion corresponds to (3.10) in the one-dimensional case. Concerning the adaptive stream phase, the flattening is made possible by a set of weights $(F^j_{\Delta\ell,\boldsymbol{\delta}})_{\boldsymbol{\delta}\in[\![-2,2]\!]^2} \subset \mathbb{R}$ such that (2.40) reads

$$\bar{\mathrm{f}}^j_{\ell,\boldsymbol{k}}(t+\Delta t) = \bar{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}}(t) + \frac{1}{2^{2\Delta\ell}}\sum_{\boldsymbol{\delta}\in[\![-2,2]\!]^2} F^j_{\Delta\ell,\boldsymbol{\delta}}\bar{\mathrm{f}}^{j,\star}_{\ell,\boldsymbol{k}+\boldsymbol{\delta}}(t). \qquad (3.12)$$

Observe that the weights $(F^j_{\Delta\ell,\boldsymbol{\delta}})_{\boldsymbol{\delta}\in[\![-2,2]\!]^2}$ for $d=2$ are not directly linked with their equivalents for $d=1$. Nevertheless, the recurrence relations they satisfy are inherited from the one-dimensional case because of the construction of the prediction operator by tensor product. Considering (3.12) from a Finite Difference point of view and expanding in Taylor series yields

$$\begin{aligned}
&\sum_{h=0}^{+\infty}\frac{\Delta t^h}{h!}\partial_t^h f^j = f^{j,\star} + \sum_{h_1=0}^{+\infty}\sum_{h_2=0}^{+\infty}\frac{2^{\Delta\ell(h_1+h_2-2)}\Delta x^{h_1+h_2}}{h_1!h_2!}\left(\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_1^{h_1}\delta_2^{h_2}F^j_{\Delta\ell,\delta_1,\delta_2}\right)\partial_{x_1}^{h_1}\partial_{x_2}^{h_2}f^{j,\star} \qquad (3.13)\\
&= \left(1+\frac{1}{2^{2\Delta\ell}}\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}F^j_{\Delta\ell,\delta_1,\delta_2}\right)f^{j,\star}\\
&+\Delta x\left(\left(\frac{1}{2^{2\Delta\ell}}\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_1F^j_{\Delta\ell,\delta_1,\delta_2}\right)\partial_{x_1}+\left(\frac{1}{2^{2\Delta\ell}}\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_2F^j_{\Delta\ell,\delta_1,\delta_2}\right)\partial_{x_2}\right)f^{j,\star}\\
&+\frac{\Delta x^2}{2}\left(\left(\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_1^2F^j_{\Delta\ell,\delta_1,\delta_2}\right)\partial_{x_1x_1}+2\left(\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_1\delta_2F^j_{\Delta\ell,\delta_1,\delta_2}\right)\partial_{x_1x_2}+\left(\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_2^2F^j_{\Delta\ell,\delta_1,\delta_2}\right)\partial_{x_2x_2}\right)f^{j,\star}\\
&+O(\Delta x^3).
\end{aligned}$$

Comparing (3.13) term-by-term with (3.11) gives the following.

Definition 3.1.2: Match

Let $d=2$. We say that the adaptive stream phase (2.40)/(3.12) matches that of the reference scheme (2.14) at order $H\in\mathbb{N}^*$ whenever

$$\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}F^j_{\Delta\ell,\delta_1,\delta_2}=0,\quad \sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2}\delta_1^{h_1}\delta_2^{h_2}F^j_{\Delta\ell,\delta_1,\delta_2}=\frac{(-\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^{h_1}(-\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^{h_2}}{2^{\Delta\ell(h_1+h_2-2)}},\quad h_1+h_2\in[\![1,H]\!]. \qquad (3.14)$$

for every level $\ell\in[\![\underline{\ell},\overline{\ell}]\!]$, thus for every $\Delta\ell\geq 0$ and for every discrete velocity $j\in[\![1,q]\!]$.

We focus on the multiresolution stream with $\gamma=1$, which is indeed the interesting case, as we have observed for $d=1$.

Proposition 3.1.5: Match for (2.14) and $\gamma=1$

Let $d=2$ and $\gamma=1$. The adaptive stream phase matches that of the reference according to Definition 3.1.2 up to order $H=3$.

Proof. The idea of the proof is to re-use the computations of Proposition 3.1.3, thanks to the construction of the prediction operator $\boldsymbol{P}_\Delta$ by tensor product. We gather the weights with the following ordering, spanning first the first axis and then the second:

$$\boldsymbol{F}^j_{\Delta\ell}:=(F^j_{\Delta\ell,-2,-2},F^j_{\Delta\ell,-1,-2},\dots,F^j_{\Delta\ell,2,-2},F^j_{\Delta\ell,-2,-1},\dots,F^j_{\Delta\ell,2,-1},\dots,F^j_{\Delta\ell,2,2})^{\mathsf{t}}\in\mathbb{R}^{25}.$$

Inside this vector $\boldsymbol{F}^j_{\Delta\ell}$, the coefficient $F^j_{\Delta\ell,\delta_1,\delta_2}$ has place $5\delta_2+\delta_1$, where in the proof, indices for vectors and matrices are allowed to take relative integer values around zero (*i.e.* the central row/column of $\boldsymbol{P}$) for notation purpose. We obtain the recurrence relation

$$\boldsymbol{F}^j_{\Delta\ell}=(\boldsymbol{P}\otimes\boldsymbol{P})\boldsymbol{F}^j_{\Delta\ell-1}. \qquad (3.15)$$

Thanks to the structure of the Kronecker product, we have $(\boldsymbol{P}\otimes\boldsymbol{P})_{pq}=P_{\lfloor p/5\rfloor,\lfloor q/5\rfloor}P_{p \bmod 5,q \bmod 5}$. We use the following Lemma, directly derived from Proposition 3.1.3.

Lemma 3.1.1

Let $q\in[\![-2,2]\!]$, then $\sum_{\delta=-2}^{\delta=+2}\delta^hP_{\delta q}=2^{1-h}q^h$ for $h\in[\![0,3]\!]$, with the notation according to which $0^0=1$.

Proof. The claim follows by direct inspection of the columns of $\boldsymbol{P}$. □

We continue the proof by recurrence over $\Delta\ell$. The claim trivially holds for $\Delta\ell = 0$. Consider now $h_1, h_2 \in [\![1,3]\!]$ and assume that the claim holds for $\Delta\ell - 1$, that is

$$\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2} \delta_1^{h_1}\delta_2^{h_2} F^j_{\Delta\ell-1,\delta_1,\delta_2} = \frac{(-\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^{h_1}(-\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^{h_2}}{2^{(\Delta\ell-1)(h_1+h_2-2)}}.$$

We then have

$$\begin{aligned}
\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2} \delta_1^{h_1}\delta_2^{h_2} F^j_{\Delta\ell,\delta_1,\delta_2} &= \sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2} \delta_1^{h_1}\delta_2^{h_2} \sum_{r=0}^{24} P_{\delta_2,\lfloor r/5\rfloor-2} P_{\delta_1,(r \bmod 5)-2} F^j_{\Delta\ell-1,(r \bmod 5)-2,\lfloor r/5\rfloor-2} \\
&= \sum_{r=0}^{24} F^j_{\Delta\ell-1,(r \bmod 5)-2,\lfloor r/5\rfloor-2} \left(\sum_{\delta_1=-2}^{+2} \delta_1^{h_1} P_{\delta_1,(r \bmod 5)-2}\right)\left(\sum_{\delta_2=-2}^{+2} \delta_2^{h_2} P_{\delta_2,\lfloor r/5\rfloor-2}\right) \\
&= \sum_{r=0}^{24} F^j_{\Delta\ell-1,(r \bmod 5)-2,\lfloor r/5\rfloor-2} \times 2^{1-h_1}((r \bmod 5)-2)^{h_1} \times 2^{1-h_2}(\lfloor r/5\rfloor-2)^{h_2} \\
&= \frac{1}{2^{h_1+h_2-2}} \sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2} \delta_1^{h_1}\delta_2^{h_2} F^j_{\Delta\ell-1,\delta_1,\delta_2} = \frac{(-\boldsymbol{c}_j\cdot\boldsymbol{e}_1)^{h_1}(-\boldsymbol{c}_j\cdot\boldsymbol{e}_2)^{h_2}}{2^{\Delta\ell(h_1+h_2-2)}}.
\end{aligned}$$

In this chain of equalities, we have used (3.15) for the first identity; a rearrangement for the second one; Lemma 3.1.1 for the third equality; a change of indices for the the fourth and finally the recurrence assumption. For the first equality in (3.14), again it is trivially satisfied for $\Delta\ell = 0$. Assume that it is true for $\Delta\ell - 1$, then we have—with the same computation:

$$\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2} F^j_{\Delta\ell,\delta_1,\delta_2} = 4\sum_{\delta_1=-2}^{+2}\sum_{\delta_2=-2}^{+2} F^j_{\Delta\ell-1,\delta_1,\delta_2} = 0,$$

by recurrence assumption, completing the proof. □

Remark 3.1.1 (Higher order matching for mixed terms)**.** *It is interesting to observe that thanks to the construction of the prediction operator by tensor product, the stream phase* (2.40) *with* $\gamma = 1$ *matches the reference scheme until order six for crossed terms between the axes, since in the proof of Proposition 3.1.5, we considered* $h_1, h_2 \in [\![1,3]\!]$*, whereas Definition 3.1.2 only requests* $h_1 + h_2 \in [\![1,3]\!]$*. Still, pure directional terms are not matched above order three as in the* $d = 1$ *case.*

To summarize, Proposition 3.1.5 shows how to generalize the study of the adaptive scheme when $d \geq 2$, to recover the same results than the study for $d = 1$ by taking advantage of the tensor product structure. Again, for most of the applications, one needs to consider $\gamma \geq 1$.

3.1.4 Numerical tests

In the previous discussion, the Taylor expansions we have performed were formal and valid for smooth solutions in the limit of small Δx_ℓ for any considered level $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$. The aim of the following numerical tests is to assess the previous approach by showing that it provides a useful tool to *a priori* study the behavior of the adaptive scheme. We shall observe that the result from the actual numerical simulations adhere to the formal expansions we have detailed in Section 3.1.1 and Section 3.1.3.

3.1.4.1 Test cases and metrics

Let us start by consider a series of test cases, associated configurations and error indicators. We test on uniform meshes which are coarsened until reaching the lowest authorized level $\underline{\ell}$, where the adaptive scheme is utilized. Indeed, the main focus of this work is not to evaluate the quality of the grid adaptation with respect to the parameter ϵ, which has been the subject of Chapter 2. Besides the fact that this new setting is relevant according to Section 3.1.1.1 and because the match properties defined in Definition 3.1.1 and Definition 3.1.2 hold uniformly in

Table 3.1: Summary of the test cases for the analysis of the stream phase without use of mesh adaptation.

d	Equation	Ref. scheme	Configuration	Test nb.
1	Linear advection eq. $\partial_t u + \partial_x(Vu) = 0$	D_1Q_2 Section 1.5.1	Fixed $\overline{\ell} - \underline{\ell}$, increasing $\overline{\ell}$	(I), Section 3.1.4.2
1	Linear advection-diffusion eq. $\partial_t u + \partial_x(Vu) - \mu\partial_{xx}u = 0$	D_1Q_3 Section 1.5.2	Fixed $\overline{\ell}$, decreasing $\overline{\ell}$	(II), Section 3.1.4.3
1	Viscous Burgers eq. $\partial_t u + \partial_x(u^2/2) - \mu\partial_{xx}u = 0$	D_1Q_3 Section 1.5.2	Fixed $\overline{\ell}$, decreasing $\overline{\ell}$	(III), Section 3.1.4.4
2	Linear advection-diffusion eq. $\partial_t u + \nabla_{\boldsymbol{x}} \cdot (\boldsymbol{V}u) - \mu\Delta_{\boldsymbol{x}}u = 0$	D_2Q_9 Section 1.5.4	Fixed $\overline{\ell}$, decreasing $\overline{\ell}$	(IV), Section 3.1.4.5

$\Delta\ell$, it also provides a worst case scenario to undoubtedly prove the resilience of our numerical strategy. This could be the case when one performs mesh adaptation yet selecting a very large threshold ϵ, so to that the smoothness of the solution allows to coarsen the grid everywhere. As a matter of fact, similar scenarios can also take place when the mesh is updated using some stiff numerical solution (for example a phase-field variable as in [Fakhari et al., 2016]) but we still want to achieve a good accuracy in the coarsely meshed areas for the non-stiff variables (for example the velocity field in the incompressible Navier-Stokes system [N'Guessan et al., 2021]).

In these tests, the leaves collision given by (2.38) is used. The summary of the four configurations that we test is given on Table 3.1: they include both the 1D and the 2D framework with linear and non-linear equations. Observe that we utilize all the numerical schemes under acoustic scaling between space and time. Yet, the previous study fits other scalings because it only pertains to the spatial part, whereas the time step is dictated by the finest resolution. All the schemes have only one conserved variable, *i.e.* $N = 1$, but all the previous study is independent of the number of conserved variables, since it pertains to the stream phase of the method. We are not interested in boundary condition, thus we enforce 0th order extrapolation boundary conditions for any test case.

When the reference scheme is expected to converge to the solution of the listed equation as $\overline{\ell} \to +\infty$, we consider a fixed number of coarsening steps $\overline{\ell} - \underline{\ell}$ and we increase the maximal level $\overline{\ell}$ to experimentally observe convergence. On the other hand, when the reference scheme is not convergent to the solution of the target equation under the scaling at hand, the maximum level $\overline{\ell}$ is fixed and the number of coarsenings $\Delta\underline{\ell} = \overline{\ell} - \underline{\ell}$ is increased. The first kind of situation aims at evaluating the possible "interference" of the adaptive strategy with the order of convergence of the reference scheme and precisely show that the theoretical analysis allows to study such a phenomenon. On the other hand, the second kind of setting aims at quantifying the effect of the adaptive scheme compared to the error of the reference scheme and shows that the previous analysis allows to construct a comparative evaluation of the various methods.

We monitor the following metrics on the conserved moment, which are all taken with respect to the L^1 norm and normalized using the norm of the exact solution. They are considered at final time T.

- E_{ref}, error of the reference scheme with respect to the exact solution of the problem. It is intrinsic to the numerical method and, depending on the specific scheme, the target model and the scaling between space and time discretization, it can converge or not as $\Delta x \to 0$.
- $E_{\text{coa}}^{\underline{\ell}}$, error of the adaptive scheme with respect to the exact solution measured at level $\underline{\ell}$.
- $E_{\text{coa}}^{\overline{\ell}}$, error of the adaptive scheme with respect to the exact solution measured at level $\overline{\ell}$, with the solution of the adaptive scheme built at $\overline{\ell}$ using the reconstruction operator.
- D_{coa}, difference between the reference and adaptive scheme, where the adaptive datum has been reconstructed at finest level $\overline{\ell}$ in order to compare it with the solution of the reference scheme. It is converging as $\Delta\underline{\ell} \to 0$.

The objective is to keep $D_{\text{coa}} \ll E_{\text{ref}}$ regardless of the fact that E_{ref} converges, so that the error of the adaptive scheme is largely dominated by that of the reference scheme. By the triangle inequality, the following control on the error of the adaptive method holds $E_{\text{coa}}^{\overline{\ell}} \leq E_{\text{ref}} + D_{\text{coa}}$: the error of the adaptive method is the result of two contributions, the error of the reference scheme (which in principle cannot be alleviated) and the difference between the behavior of the adaptive and the reference scheme (to be alleviated by increasing γ).

Table 3.2: Test (I) for the 1D linear advection equation taking $\Delta\underline{\ell} = 2$ and $s_2 = 1$. Numerical convergence rates are reported between parenthesis.

		(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\overline{\ell}$	E_{ref}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}
3	1.20e+0	1.21e+0	1.09e+0	8.92e-1	8.41e-1	1.06e+0	8.31e-1	8.58e-1	1.07e+0	8.40e-1
4	1.01e+0 (0.25)	1.43e+0	1.41e+0	7.04e-1 (0.34)	1.07e+0	1.16e+0	2.30e-1 (1.85)	1.13e+0	1.23e+0	3.95e-1 (1.09)
5	7.93e-1 (0.35)	1.24e+0	1.29e+0	6.73e-1 (0.06)	9.27e-1	8.89e-1	1.12e-1 (1.04)	9.55e-1	9.26e-1	2.66e-1 (0.57)
6	5.67e-1 (0.48)	1.08e+0	1.09e+0	6.22e-1 (0.11)	6.21e-1	6.07e-1	5.02e-2 (1.16)	6.19e-1	6.11e-1	1.48e-1 (0.84)
7	3.71e-1 (0.61)	8.68e-1	8.66e-1	5.41e-1 (0.20)	3.86e-1	3.83e-1	1.67e-2 (1.59)	3.81e-1	3.80e-1	7.16e-2 (1.05)
8	2.22e-1 (0.74)	6.37e-1	6.37e-1	4.31e-1 (0.33)	2.26e-1	2.25e-1	4.02e-3 (2.05)	2.24e-1	2.24e-1	2.88e-2 (1.31)
9	1.24e-1 (0.84)	4.29e-1	4.29e-1	3.09e-1 (0.48)	1.25e-1	1.25e-1	7.26e-4 (2.47)	1.25e-1	1.24e-1	9.79e-3 (1.56)
10	6.61e-2 (0.91)	2.64e-1	2.64e-1	1.99e-1 (0.64)	6.62e-2	6.62e-2	1.04e-4 (2.80)	6.62e-2	6.61e-2	2.93e-3 (1.74)
11	3.42e-2 (0.95)	1.51e-1	1.51e-1	1.17e-1 (0.77)	3.42e-2	3.42e-2	1.24e-5 (3.06)	3.42e-2	3.42e-2	8.10e-4 (1.86)
12	1.74e-2 (0.97)	8.13e-2	8.13e-2	6.39e-2 (0.87)	1.74e-2	1.74e-2	2.27e-6 (2.46)	1.74e-2	1.74e-2	2.13e-4 (1.92)

Table 3.3: Test (I) for the 1D linear advection equation taking $\Delta\underline{\ell} = 2$ and $s_2 = 2$. Numerical convergence rates are reported between parenthesis.

		(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\overline{\ell}$	E_{ref}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}
3	1.27e+0	1.15e+0	1.08e+0	9.96e-1	9.20e-1	1.09e+0	8.29e-1	8.56e-1	1.08e+0	9.70e-1
4	7.53e-1 (0.75)	1.37e+0	1.37e+0	1.07e+0 (-0.10)	9.20e-1	1.09e+0	8.29e-1 (0.00)	1.22e+0	1.36e+0	9.59e-1 (0.02)
5	2.03e-1 (1.89)	1.20e+0	1.23e+0	1.20e+0 (-0.16)	7.54e-1	7.09e-1	6.64e-1 (0.32)	1.09e+0	1.07e+0	9.85e-1 (-0.04)
6	5.06e-2 (2.01)	1.01e+0	1.02e+0	1.02e+0 (0.24)	3.00e-1	3.05e-1	2.96e-1 (1.17)	6.13e-1	6.09e-1	5.84e-1 (0.75)
7	1.27e-2 (2.00)	7.91e-1	7.92e-1	7.91e-1 (0.36)	7.48e-2	7.27e-2	6.98e-2 (2.09)	2.23e-1	2.22e-1	2.11e-1 (1.47)
8	3.17e-3 (2.00)	5.67e-1	5.67e-1	5.67e-1 (0.48)	1.17e-2	1.12e-2	1.03e-2 (2.77)	5.99e-2	5.98e-2	5.67e-2 (1.90)
9	7.92e-4 (2.00)	3.71e-1	3.71e-1	3.71e-1 (0.61)	1.68e-3	1.54e-3	1.25e-3 (3.03)	1.52e-2	1.52e-2	1.44e-2 (1.98)
10	1.98e-4 (2.00)	2.22e-1	2.22e-1	2.22e-1 (0.74)	2.75e-4	2.45e-4	1.41e-4 (3.15)	3.80e-3	3.80e-3	3.60e-3 (2.00)
11	4.95e-5 (2.00)	1.24e-1	1.24e-1	1.24e-1 (0.84)	5.57e-5	5.09e-5	1.46e-5 (3.27)	9.49e-4	9.49e-4	9.00e-4 (2.00)
12	1.24e-5 (2.00)	6.61e-2	6.61e-2	6.61e-2 (0.91)	1.29e-5	1.23e-5	2.34e-6 (2.64)	2.37e-4	2.37e-4	2.25e-4 (2.00)

3.1.4.2 D_1Q_2 for the linear advection equation

The aim of this test case is to check our analysis in a case where, on the one hand, we know that the reference scheme converges as $\Delta x \to 0$ because the relaxation parameter is taken independent from Δx because no viscous effect need to be modeled and, on the other hand, the equilibria are linear thus the collision strategy does not alter the quality of the method. We expect that all the tested methods match enough terms in order to preserve the convergence of the reference method. However, two interwoven phenomena can take place. The first is a modification of the convergence rate because of D_{coa}. Second, at some point, the term D_{coa} can become non-negligible with respect to E_{ref}.

3.1.4.2.1 The problem and the scheme The target problem is the linear advection equation with constant velocity $V \in \mathbb{R}$ with exact solution

$$u(t,x) = \frac{1}{\sqrt{4\pi\mu t^\circ}} \exp\left(-\frac{(x - Vt)^2}{4\mu t^\circ}\right). \tag{3.16}$$

In this test, we shall use a final time $T = 1$, $t^\circ = 1$ and $\mu = 5\text{e}-3$, giving the initial spreading of the Gaussian, and $V = 1/2$. The numerical scheme is the D_1Q_2 detailed in Section 2.8.1.2 on a bounded domain $\Omega = [-3, 3]$. It has been theoretically shown [Dellacherie, 2014], conditionally in the L^∞ norm and unconditionally in the L^2 norm that the scheme converges—under the CFL condition $|V|/\lambda \leq 1$—linearly towards the solution (3.16) when the relaxation parameter $s_2 \in]0, 2[$ and quadratically for $s_2 = 2$ when $\Delta x \to 0$. In this test, we take the lattice velocity as $\lambda = 1$.

3.1.4.2.2 Results and discussion The test is conducted fixing the difference between the maximum level $\overline{\ell}$ and the minimum level $\underline{\ell}$, at which we perform the computation, either at $\Delta\underline{\ell} = 2$ or $\Delta\underline{\ell} = 6$ for two different relaxation parameters, namely $s_2 = 1$ and $s_2 = 2$. We progressively increase the maximum level $\overline{\ell}$ to observe the convergence of the reference scheme towards the solution and to study how the adaptive schemes behave in such a situation.

The results are provided in Table 3.2 and Table 3.3 for $\Delta\underline{\ell} = 2$ and in Table 3.4 and Table 3.5 for $\Delta\underline{\ell} = 6$. In all of them, we observe the expected behavior of the reference scheme, converging linearly for $s_2 = 1$ and quadratically for $s_2 = 2$. The errors are also presented differently in Figure 3.2 and Figure 3.3. Let us comment on the behavior of each adaptive strategy:

Table 3.4: Test (I) for the 1D linear advection equation taking $\Delta\underline{\ell} = 6$ and $s_2 = 1$. Numerical convergence rates are reported between parenthesis.

		(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\overline{\ell}$	E_{ref}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}
7	3.71e-1	1.20e+0	1.08e+0	1.05e+0	7.52e-1	1.07e+0	1.00e+0	1.22e+0	1.14e+0	1.08e+0
8	2.22e-1 (0.74)	1.44e+0	1.41e+0	1.31e+0 (-0.32)	1.03e+0	1.28e+0	1.09e+0 (-0.12)	1.83e+0	1.95e+0	1.79e+0 (-0.73)
9	1.24e-1 (0.84)	1.25e+0	1.30e+0	1.23e+0 (0.10)	7.71e-1	7.37e-1	6.23e-1 (0.81)	1.62e+0	1.65e+0	1.55e+0 (0.21)
10	6.61e-2 (0.91)	1.10e+0	1.11e+0	1.06e+0 (0.21)	2.12e-1	2.03e-1	1.53e-1 (2.03)	7.59e-1	7.61e-1	7.13e-1 (1.12)
11	3.42e-2 (0.95)	8.88e-1	8.85e-1	8.58e-1 (0.30)	4.71e-2	4.47e-2	1.89e-2 (3.01)	2.34e-1	2.33e-1	2.18e-1 (1.71)
12	1.74e-2 (0.97)	6.57e-1	6.57e-1	6.41e-1 (0.42)	1.90e-2	1.80e-2	1.94e-3 (3.28)	6.17e-2	6.16e-2	5.79e-2 (1.91)

Table 3.5: Test (I) for the 1D linear advection equation taking $\Delta\underline{\ell} = 6$ and $s_2 = 2$. Numerical convergence rates are reported between parenthesis.

		(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\overline{\ell}$	E_{ref}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}
7	1.27e-2	1.20e+0	1.08e+0	1.08e+0	7.47e-1	1.07e+0	1.07e+0	1.22e+0	1.14e+0	1.14e+0
8	3.17e-3 (2.00)	1.44e+0	1.41e+0	1.41e+0 (-0.38)	1.05e+0	1.31e+0	1.31e+0 (-0.29)	1.86e+0	1.98e+0	1.98e+0 (-0.80)
9	7.92e-4 (2.00)	1.25e+0	1.30e+0	1.30e+0 (0.12)	7.84e-1	7.65e-1	7.65e-1 (0.77)	1.68e+0	1.72e+0	1.72e+0 (0.20)
10	1.98e-4 (2.00)	1.10e+0	1.10e+0	1.10e+0 (0.23)	2.00e-1	1.94e-1	1.94e-1 (1.98)	8.05e-1	8.07e-1	8.07e-1 (1.09)
11	4.95e-5 (2.00)	8.84e-1	8.82e-1	8.82e-1 (0.32)	2.33e-2	2.22e-2	2.22e-2 (3.13)	2.41e-1	2.41e-1	2.41e-1 (1.74)
12	1.24e-5 (2.00)	6.53e-1	6.53e-1	6.53e-1 (0.43)	2.57e-3	2.12e-3	2.12e-3 (3.39)	6.09e-2	6.11e-2	6.11e-2 (1.98)

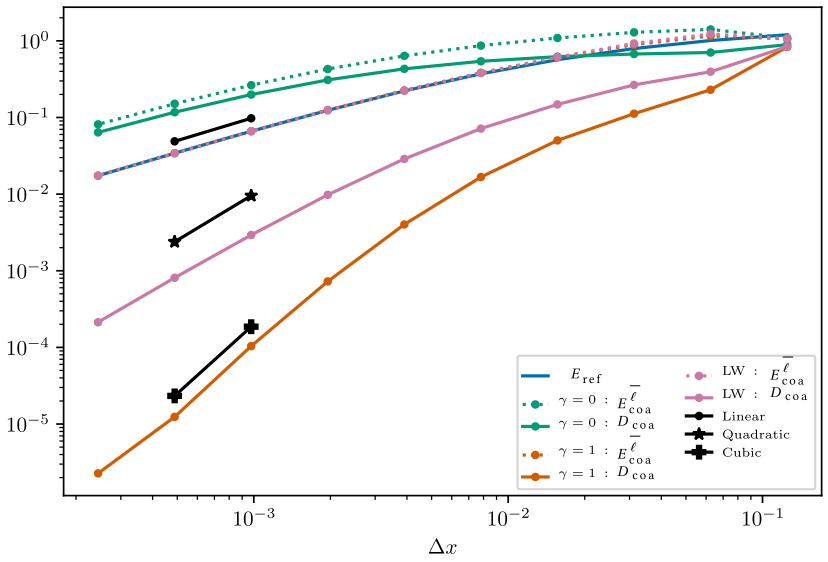


Figure 3.2: Test (I) for the 1D linear advection equation taking $\Delta\underline{\ell} = 2$ and $s_2 = 1$. As expected, $D_{\text{coa}} = O(\Delta x)$ for $\gamma = 0$, $D_{\text{coa}} = O(\Delta x^3)$ for $\gamma = 1$ and $D_{\text{coa}} = O(\Delta x^2)$ for Lax-Wendroff. All $E^{\overline{\ell}}_{\text{coa}} = O(\Delta x)$.

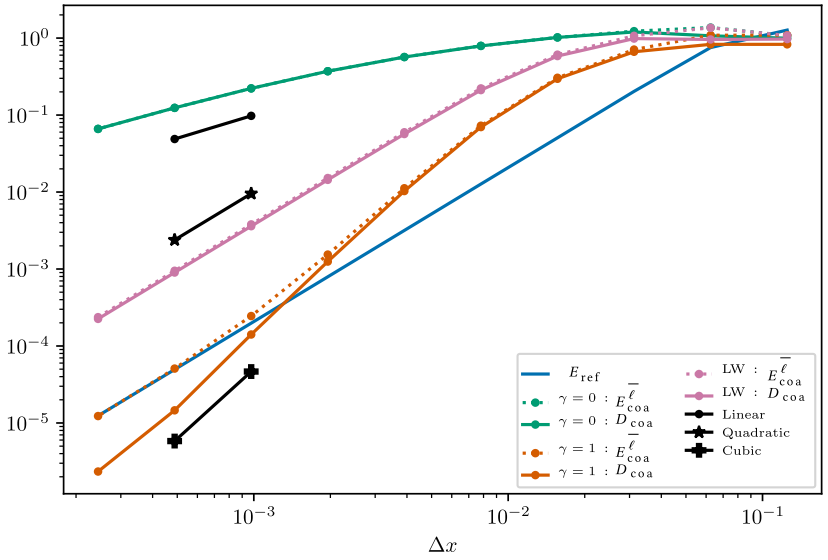


Figure 3.3: Test (I) for the 1D linear advection equation taking $\Delta\underline{\ell} = 2$ and $s_2 = 2$. As expected, $D_{\text{coa}} = O(\Delta x)$ for $\gamma = 0$, $D_{\text{coa}} = O(\Delta x^3)$ for $\gamma = 1$ and $D_{\text{coa}} = O(\Delta x^2)$ for Lax-Wendroff. All $E^{\overline{\ell}}_{\text{coa}} = O(\Delta x)$.

- Multiresolution scheme (2.40) for $\gamma = 0$. In the case where the reference method is first-order convergent ($s_2 = 1$), we observe that $E^{\overline{\ell}}_{\text{coa}}$ is also first-order convergent but, especially for $\Delta\underline{\ell} = 6$, D_{coa} dominates against E_{ref}, which is the reason why the green full line and the green dashed lines are not superposed in Figure 3.2. Moreover, this is the reason why for $s_2 = 2$, despite the fact that E_{ref} converges quadratically, $E^{\overline{\ell}}_{\text{coa}}$ converges only linearly due to the limitations imposed by the convergence ratio of D_{coa}.
- Multiresolution scheme (2.40) for $\gamma = 1$. We observe that in any case the convergence rate of D_{coa} is third-order. Therefore, we always obtain the same convergence rate of E_{ref} for $E^{\overline{\ell}}_{\text{coa}}$.
- Lax-Wendroff scheme (3.2). Since the convergence rate of D_{coa} is second-order, we always observe no alteration of the convergence rate of the reference scheme by the adaptive method.

Finally, observe that for any method the convergence rates of D_{coa} are erratic for small $\overline{\ell}$ presumably because Taylor expansions are not fully legitimate for coarse resolutions. These numerical experiments confirm the theoretical study and show that the adaptive method should match enough terms to avoid alterations of the convergence rates of the reference scheme. In particular, if the reference method converges at order H, the adaptive stream phase must match at least at order H as well, according to Definition 3.1.1.

In this particular case, we propose the outcome of the equivalent equation analysis by [Dubois, 2008] for illustrative purpose. For the reference scheme, we obtain

$$\partial_t m^1 + V\partial_x m^1 = \lambda\Delta x\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(1 - \frac{V^2}{\lambda^2}\right)\partial_{xx} m^1 - \frac{\lambda\Delta x^2}{6}\frac{V}{\lambda}\left(1 - \frac{V^2}{\lambda^2}\right)\partial_x^3 m^1 + \frac{\lambda\Delta x^3}{12}\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(1 - \frac{V^4}{\lambda^4}\right)\partial_x^4 m^1 + O(\Delta x^4).$$

- Multiresolution scheme (2.40) for $\gamma = 0$. We obtain the modified equation:

$$\partial_t m^1 + V\partial_x m^1 = \lambda\Delta x\left(\frac{1}{2}(2^{\Delta\ell} - 1) + \left(\frac{1}{s_2} - \frac{1}{2}\right)\left(1 - \frac{V^2}{\lambda^2}\right)\right)\partial_{xx} m^1 + O(\Delta x^2).$$

  We see that the fact of increasing the level jump $\Delta\ell$ adds numerical diffusion, proportionally to $2^{\Delta\ell}\Delta x = \Delta x_\ell$. This was predicted by the previous theory.

- Multiresolution scheme (2.40) for $\gamma = 1$. It is possible to compute the powers of the matrix $\boldsymbol{P}$ given in Proposition 3.1.3 using symbolic computations. This yields the following weights for the velocity indexed by $j = 1$:

$$F^1_{\Delta\ell,-2} = \frac{2^{-2\Delta\ell}}{24}(-3\times 2^{\Delta\ell}\Delta\ell - 2^{2\Delta\ell+1} + 2), \quad F^1_{\Delta\ell,-1} = -\frac{2^{-2\Delta\ell}}{6}(-3\times 2^{\Delta\ell}\Delta\ell - 3\times 2^{\Delta\ell} - 2^{2\Delta\ell+2} + 1),$$
$$F^1_{\Delta\ell,0} = \frac{2^{-\Delta\ell}}{4}(-3\Delta\ell + 2^{2\Delta\ell+2} - 4) - 2^{\Delta\ell},$$
$$F^1_{\Delta\ell,1} = \frac{2^{-2\Delta\ell}}{6}(3\times 2^{\Delta\ell}\Delta\ell + 3\times 2^{\Delta\ell} - 2^{2\Delta\ell+2} + 1), \quad F^1_{\Delta\ell,2} = -\frac{2^{-2\Delta\ell}}{24}(3\times 2^{\Delta\ell}\Delta\ell - 2^{2\Delta\ell+1} + 2).$$

  By this, we come to the equivalent equation of the scheme, which reads

$$\begin{aligned}\partial_t m^1 + V\partial_x m^1 =& \lambda\Delta x\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(1 - \frac{V^2}{\lambda^2}\right)\partial_{xx} m^1 - \frac{\lambda\Delta x^2}{6}\frac{V}{\lambda}\left(1 - \frac{V^2}{\lambda^2}\right)\partial^3_x m^1 \\ &+ \frac{\lambda\Delta x^3}{24}\left((2^{\Delta\ell}(1-3\Delta\ell) - 1) + 2\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(1 - \frac{V^4}{\lambda^4}\right)\right)\partial^4_x m^1 + O(\Delta x^4).\end{aligned}$$

  We observe that the perturbation in $\Delta\ell$ appears only at third order with a bi-Laplacian term, as predicted.

- Lax-Wendroff scheme (3.2). We obtain the equivalent equation

$$\partial_t m^1 + V\partial_x m^1 = \lambda\Delta x\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(1 - \frac{V^2}{\lambda^2}\right)\partial_{xx} m^1 - \frac{\lambda\Delta x^2}{6}\frac{V}{\lambda}\left((2^{2\Delta\ell} - 1) + \left(1 - \frac{V^2}{\lambda^2}\right)\right)\partial^3_x m^1 + O(\Delta x^3).$$

  We observe that the perturbation due to $\Delta\ell$ appears at second order, increasing the dispersion on the scheme, as predicted.

3.1.4.3 D_1Q_3 for the linear advection-diffusion equation

In this test case, the reference method is no longer convergent under acoustic scaling because the diffusion term arises as numerical diffusion—thus asymptotically vanishes—and the relaxation parameters are adjusted to recover the right diffusion terms but would asymptotically reach forbidden values, see [Boghosian et al., 2018]. Despite this lack of convergence, the aim of the simulations is to show that the adaptive scheme for $\gamma = 0$ is not accurate enough to reproduce the physics of the reference algorithm. Moreover, we shall observe that the Lax-Wendroff scheme correctly accounts for the diffusion terms but can introduce spurious oscillations due to the lack of matching of the third-order terms. This is not the case for the multiresolution scheme with $\gamma = 1$.

3.1.4.3.1 The problem and the scheme The target problem is the linear advection-diffusion equation with constant velocity $V \in \mathbb{R}$ and diffusion coefficient $\mu > 0$, with exact solution

$$u(t,x) = \frac{1}{\sqrt{4\pi\mu(t^\circ + t)}}\exp\left(-\frac{(x - Vt)^2}{4\mu(t^\circ + t)}\right). \tag{3.17}$$

The parameters of the problem are the same as Section 3.1.4.2. The numerical scheme that we employ is the D_1Q_3 detailed in Section 1.5.2 using the choice of moments given by (1.5) on a bounded domain $\Omega = [-3,3]$. We take $\lambda = 1$ and—in order to obtain the right transport velocity and diffusivity—the equilibria are taken as $m^{\mathrm{eq}}_2(\mathsf{m}_1) = V\mathsf{m}_1$ and $m^{\mathrm{eq}}_3(\mathsf{m}_1) = \mathsf{m}_1$ with relaxation parameters $s_3 = 1$ and

$$s_2 = \left(\frac{1}{2} + \frac{\lambda\mu}{\Delta x(1 - V^2)}\right)^{-1}.$$

According to the discussion by [Boghosian et al., 2018], the scheme is in general not convergent towards the solution of the advection-diffusion equation, because $s_2 \to 0$ as $\Delta x \to 0$. Still, it can be used in an intermediate regime where Δx is not too small, thus s_2 is sufficiently away from 0.

Table 3.6: Test (II) for the 1D linear advection-diffusion equation taking $\overline{\ell} = 11$ and performing the computation using a mesh at level $\underline{\ell}$ as indicated.

	(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\Delta\underline{\ell}$	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}
0	1.94e-2	1.94e-2	0.00e+0	1.94e-2	1.94e-2	0.00e+0	1.94e-2	1.94e-2	0.00e+0
1	2.30e-2	2.30e-2	1.55e-2	1.94e-2	1.94e-2	7.88e-7	1.94e-2	1.94e-2	3.63e-5
2	4.68e-2	4.68e-2	4.52e-2	1.94e-2	1.94e-2	3.41e-6	1.92e-2	1.92e-2	1.82e-4
3	9.92e-2	9.92e-2	9.94e-2	1.94e-2	1.94e-2	1.31e-5	1.87e-2	1.87e-2	7.63e-4
4	1.91e-1	1.91e-1	1.92e-1	1.94e-2	1.94e-2	5.40e-5	1.65e-2	1.65e-2	3.09e-3
5	3.33e-1	3.33e-1	3.34e-1	1.93e-2	1.93e-2	2.78e-4	8.18e-3	8.32e-3	1.24e-2
6	5.25e-1	5.24e-1	5.26e-1	1.84e-2	1.84e-2	1.74e-3	3.11e-2	3.16e-2	5.03e-2
7	7.51e-1	7.47e-1	7.48e-1	1.10e-2	1.07e-2	1.89e-2	1.93e-1	1.96e-1	2.15e-1

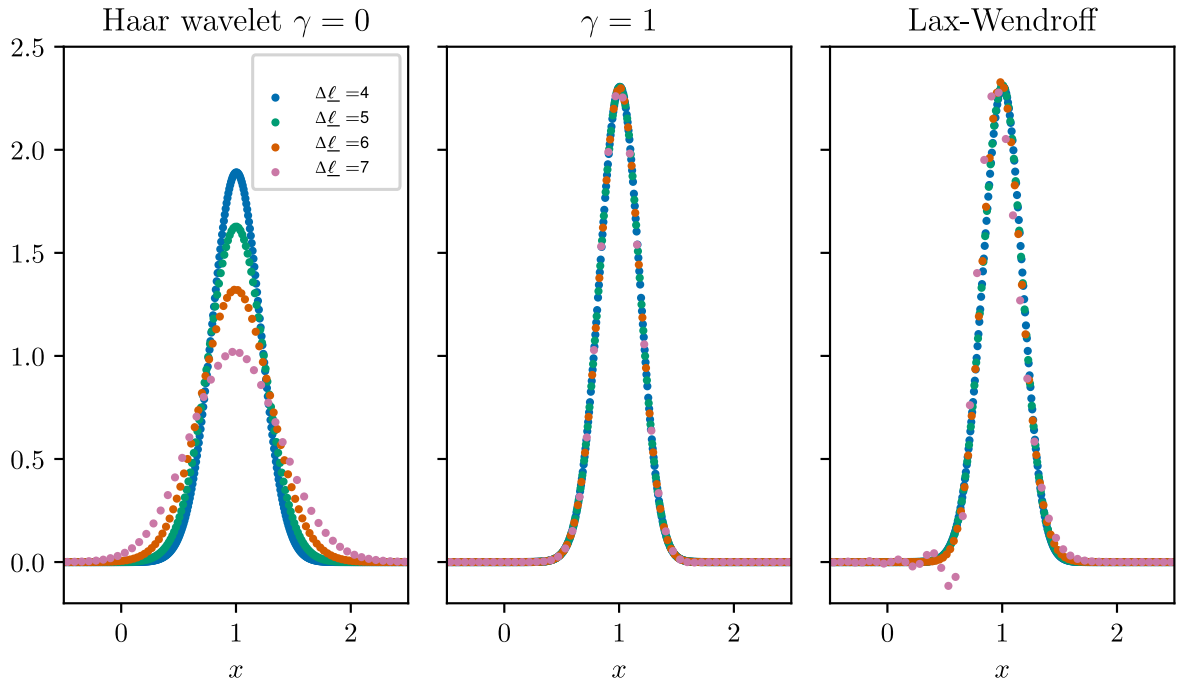


Figure 3.4: Test (II) for the 1D linear advection-diffusion equation. Solution at final time T shown on a sector of the domain for different $\Delta\underline{\ell}$ and stream schemes. We observe the unmatched dissipation for $\gamma = 0$, excellent agreement for $\gamma = 1$ and good agreement with spurious oscillations for Lax-Wendroff.

3.1.4.3.2 Results and discussion Since we are no longer interested in convergence, we take a fixed maximum level $\overline{\ell} = 11$ and we vary the minimum level $\underline{\ell}$ at which we perform computations. The results are given on Table 3.6 and the solution at final time T for some $\Delta\underline{\ell}$ is shown in Figure 3.4. We remark that:

- Multiresolution scheme (2.40) for $\gamma = 0$. The inertial term pertaining to the advection velocity V is correctly represented in accordance with Proposition 3.1.2 for any $\Delta\underline{\ell}$, because the packet is transported at the right velocity until reaching the point $x = 1$. Nevertheless, the dissipative term is not correct because we have an excess of numerical diffusion as long as $\Delta\underline{\ell}$ grows. This was predicted by the theoretical framework. Even if the dispersive term is not matched either, this does not affect the stability of the method because of the large amount of available numerical diffusion. As expected, D_{coa} is not negligible compared to E_{ref} for any $\Delta\underline{\ell} \geq 1$.
- Multiresolution scheme (2.40) for $\gamma = 1$. We observe that both the inertial and dissipative terms are correctly represented. Moreover, according to our intuition, since the dispersive terms are untouched compared to the target expansion, *cf.* Proposition 3.1.3, the adaptive method remains stable and does not produce spurious oscillations when the reference method is stable. Looking at the errors more carefully, we see that the additional error D_{coa} is negligible compared to E_{ref} for $\Delta\underline{\ell} < 6$ or 7.

Table 3.7: Test (III) for the 1D viscous Burgers equation taking $\overline{\ell} = 11$ and $\mu = 5e{-}2$ (small diffusion) and performing the computation using a mesh at level $\underline{\ell}$ as indicated.

	(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\Delta\underline{\ell}$	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}	$E^{\underline{\ell}}_{\text{coa}}$	$E^{\overline{\ell}}_{\text{coa}}$	D_{coa}
0	1.94e-2	1.94e-2	0.00e+0	1.94e-2	1.94e-2	0.00e+0	1.94e-2	1.94e-2	0.00e+0
1	2.30e-2	2.30e-2	1.55e-2	1.94e-2	1.94e-2	7.88e-7	1.94e-2	1.94e-2	3.63e-5
2	4.68e-2	4.68e-2	4.52e-2	1.94e-2	1.94e-2	3.41e-6	1.92e-2	1.92e-2	1.82e-4
3	9.92e-2	9.92e-2	9.94e-2	1.94e-2	1.94e-2	1.31e-5	1.87e-2	1.87e-2	7.63e-4
4	1.91e-1	1.91e-1	1.92e-1	1.94e-2	1.94e-2	5.40e-5	1.65e-2	1.65e-2	3.09e-3
5	3.33e-1	3.33e-1	3.34e-1	1.93e-2	1.93e-2	2.78e-4	8.18e-3	8.32e-3	1.24e-2
6	5.25e-1	5.24e-1	5.26e-1	1.84e-2	1.84e-2	1.74e-3	3.11e-2	3.16e-2	5.03e-2
7	7.51e-1	7.47e-1	7.48e-1	1.10e-2	1.07e-2	1.89e-2	1.93e-1	1.96e-1	2.15e-1

- Lax-Wendroff scheme (3.2). This scheme correctly matches the inertial and dissipative phenomena. Nevertheless, as $\Delta\underline{\ell}$ grows, we observe the formation of spurious oscillations and the packet is not perfectly centered at $x = 1$. This is presumably due to the modification of the third-order dispersion as theoretically observed. For the tested case, this is not enough to induce instabilities. This shows that this adaptive method can be subjected to instabilities even when the reference method is stable because of the modification of the dispersion. The additional error D_{coa} is negligible compared to E_{ref} for $\Delta\underline{\ell} < 3$ or 4.

Once again, the theoretical analysis is fully corroborated by the numerical behavior of the schemes and show that, even in this quite simple framework, the multiresolution schemes for $\gamma \geq 1$ are the most reliable ones.

3.1.4.4 D_1Q_3 for the viscous Burgers equation

We now turn to a non-linear problem. In this case, the choice of model for the collision phase between (2.38) and (2.37) is no longer negligible. For the moment, we aim at proving that our previous analysis is still meaningful in this context upon verification of the smoothness assumption. Indeed, we see that in the case of singularities the previous analysis is no longer well-grounded due to the lack of smoothness and we thus understand the strong interest of dynamic mesh adaptation using multiresolution.

3.1.4.4.1 The problem and the scheme We consider the solution of the viscous Burgers equation with viscosity μ, given by

$$u(t,x) = \sqrt{\frac{4\mu}{t}} \frac{\int_{-\infty}^{+\infty} \eta \exp\Big(-\frac{1}{4\mu}\operatorname{erf}\Big(\frac{x}{\sqrt{4\mu t^\circ}} - \sqrt{\frac{t}{t^\circ}}\eta\Big)\Big) e^{-\eta^2} \mathrm{d}\eta}{\int_{-\infty}^{+\infty} \exp\Big(-\frac{1}{4\mu}\operatorname{erf}\Big(\frac{x}{\sqrt{4\mu t^\circ}} - \sqrt{\frac{t}{t^\circ}}\eta\Big)\Big) e^{-\eta^2} \mathrm{d}\eta},$$

where the solution has been obtained following [Landajuela, 2011] and the integrals with weights $e^{-\eta^2}$ shall be approximated with Gauss-Hermite formulæwith 100 quadrature points. We consider a final time $T = 1$, $t^\circ = 1$ and either $\mu = 5e-2$ (large diffusion) or $\mu = 5e-3$ (small diffusion). On the domain $\Omega = [-3, 3]$, we use the same D_1Q_3 as Section 3.1.4.3 with lattice velocity $\lambda = 4$ and $m_2^{\text{eq}}(\mathsf{m}_1) = (\mathsf{m}_1)^2/2$ and $s_3 = 1$ with

$$m_3^{\text{eq}}(\mathsf{m}_1) = \frac{(\mathsf{m}_1)^3}{3} + 4\mathsf{m}_1, \qquad s_2 = \left(\frac{1}{2} + \frac{\lambda\mu}{4\Delta x}\right)^{-1}, \qquad \text{(large diffusion)},$$
$$m_3^{\text{eq}}(\mathsf{m}_1) = \frac{(\mathsf{m}_1)^3}{3} + \mathsf{m}_1, \qquad s_2 = \left(\frac{1}{2} + \frac{\lambda\mu}{\Delta x}\right)^{-1}, \qquad \text{(small diffusion)}.$$

3.1.4.4.2 Results and discussion We first perform the same kind of test as Section 3.1.4.3 and the results are on Table 3.7, Table 3.8, Figure 3.5 and Figure 3.6. We point out the following facts:

- Multiresolution scheme (2.40) for $\gamma = 0$. In the case of large diffusion, we see that the method adds much numerical diffusion yielding unreliable results. On the other hand, for the small diffusion, the result seems good from a graphic point of view because of the secondary role of diffusion on the shape of the solution. However, the discrepancies from the reference scheme are important in both cases. Compared to the other strategies, the difference with respect to the reference algorithm is larger, as expected: starting from $\Delta\underline{\ell} = 3$ (for large diffusion) and $\Delta\underline{\ell} = 2$ (for small diffusion), the term $\Delta\ell$ can no longer be neglected.

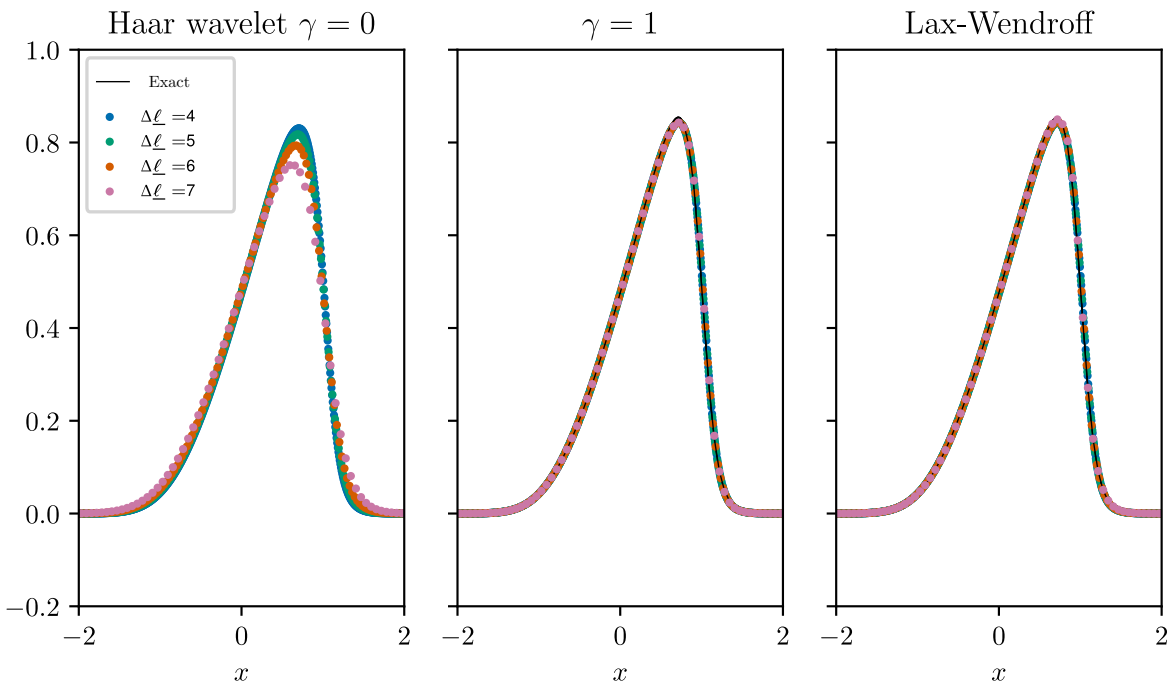


Figure 3.5: Test (III) for the 1D viscous Burgers equation with $\mu = 5e-2$ (large diffusion). Solution at final time T shown on a sector of the domain for different $\Delta\underline{\ell}$ and stream schemes. We remark the wrong diffusivity for $\gamma = 0$ and reliable results for $\gamma = 1$ and Lax-Wendroff. Exact solution plotted with a black solid line.

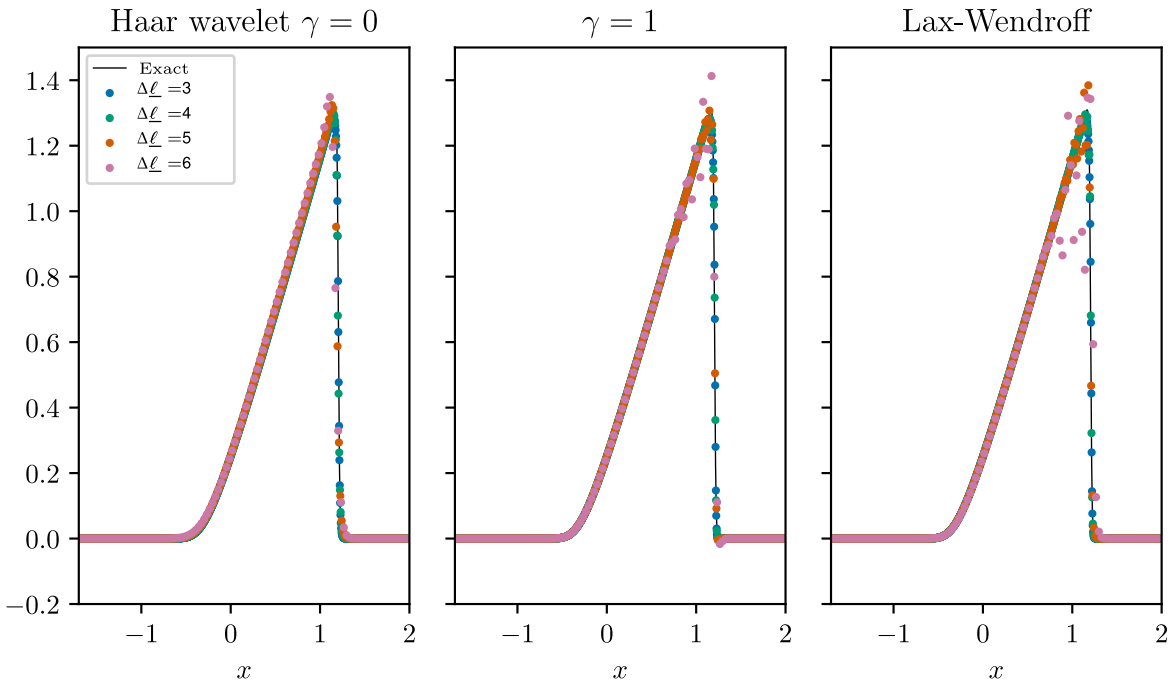


Figure 3.6: Test (III) for the 1D viscous Burgers equation with $\mu = 5e-3$ (small diffusion). Solution at final time T shown on a sector of the domain for different $\Delta\underline{\ell}$ and stream schemes. We remark the wrong diffusivity for $\gamma = 0$ and reliable results for $\gamma = 1$ and Lax-Wendroff. Exact solution plotted with a black solid line.

Table 3.8: Test (III) for the 1D viscous Burgers equation taking $\overline{\ell} = 11$ and $\mu = 5\text{e}{-3}$ (small diffusion) and performing the computation using a mesh at level $\underline{\ell}$ as indicated.

	(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\Delta\underline{\ell}$	$E_{\text{coa}}^{\underline{\ell}}$	$E_{\text{coa}}^{\overline{\ell}}$	D_{coa}	$E_{\text{coa}}^{\underline{\ell}}$	$E_{\text{coa}}^{\overline{\ell}}$	D_{coa}	$E_{\text{coa}}^{\underline{\ell}}$	$E_{\text{coa}}^{\overline{\ell}}$	D_{coa}
0	1.94e-2	1.94e-2	0.00e+0	1.94e-2	1.94e-2	0.00e+0	1.94e-2	1.94e-2	0.00e+0
1	2.30e-2	2.30e-2	1.55e-2	1.94e-2	1.94e-2	7.88e-7	1.94e-2	1.94e-2	3.63e-5
2	4.68e-2	4.68e-2	4.52e-2	1.94e-2	1.94e-2	3.41e-6	1.92e-2	1.92e-2	1.82e-4
3	9.92e-2	9.92e-2	9.94e-2	1.94e-2	1.94e-2	1.31e-5	1.87e-2	1.87e-2	7.63e-4
4	1.91e-1	1.91e-1	1.92e-1	1.94e-2	1.94e-2	5.40e-5	1.65e-2	1.65e-2	3.09e-3
5	3.33e-1	3.33e-1	3.34e-1	1.93e-2	1.93e-2	2.78e-4	8.18e-3	8.32e-3	1.24e-2
6	5.25e-1	5.24e-1	5.26e-1	1.84e-2	1.84e-2	1.74e-3	3.11e-2	3.16e-2	5.03e-2
7	7.51e-1	7.47e-1	7.48e-1	1.10e-2	1.07e-2	1.89e-2	1.93e-1	1.96e-1	2.15e-1

- Multiresolution scheme (2.40) for $\gamma = 1$. The plots of the solution show that it agrees well with the expected one. We can notice a slight crushing of the solution for the large diffusion which can be considered a fourth order effect. In the case of small diffusion, despite the fact that the reference scheme does not oscillate close to the steep zone of the solution, we see that the adaptive method does so for large $\Delta\underline{\ell}$. This cannot be due to third-order terms, since they are matched, so one may argue that these are fifth-order effects (not likely) or the consequence of the fact that we are no longer allowed to perform Taylor expansions either because the spatial step is too large or because the solution is not smooth enough. Indeed, as already said, this is the proof that using a fixed coarsened mesh is not a good approach to deal with moving singularities. This context calls for dynamically adapted meshes and error control. Still, the difference with the reference scheme is minimized for this choice of reconstruction and the impact of the adaptive scheme can be neglected for any $\Delta\underline{\ell}$ for the large diffusion and until $\Delta\underline{\ell}$ for the small diffusion.
- Lax-Wendroff scheme (3.2). The result for the large diffusion seems reliable, even if the method slightly "overshoots" the exact solution presumably due to the third-order mismatch. On the other hand, the test with small diffusion clearly shows that the method perturbs the dispersive terms at third order, inducing oscillations near the kinky zones of the solution. Compared to the reference solution, the behavior of the Lax-Wendroff scheme is situated half-way between those for $\gamma = 0$ and $\gamma = 1$, as expected. in particular, the impact of the adaptive stream is negligible until $\Delta\underline{\ell} = 7$ (for large diffusion) and $\Delta\underline{\ell} = 3$ (for small diffusion).

For each stream strategy, we see that D_{coa} stops to be negligible compared E_{ref} earlier for the small diffusion than for the large. This is coherent with the fact that the solution develops more high-frequency modes. Moreover, one limiting factor of the theoretical analysis are the implicit smoothness assumptions on the solutions. In the case where the solution is close to singular and especially for large $\Delta\underline{\ell}$, the behavior of the numerical scheme on a uniform coarsened mesh deviates from the theoretical predictions because the smoothness assumption is not valid. From a multiresolution perspective, the lack of smoothness translates into the fact that the details of the solution at the finest level $\overline{\ell}$ are not small close to the blowup.

3.1.4.5 D_2Q_9 for the linear advection-diffusion equation

To corroborate the extension of the previous analysis to the multidimensional setting done in Section 3.1.3. We selected a quite "rich" numerical model in terms of degrees of freedom to show the generality of our analysis.

3.1.4.5.1 The problem and the scheme We consider the linear advection-diffusion equation with constant velocity $\boldsymbol{V} \in \mathbb{R}^2$ and diffusion coefficient $\mu > 0$, with exact solution

$$u(t, \boldsymbol{x}) = \frac{1}{\sqrt{4\pi\mu(t^\circ + t)}} \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{V}t\|_2^2}{4\mu(t^\circ + t)}\right). \tag{3.18}$$

In the test, we employ $T = 1/2$, $t^\circ = 1$, $\boldsymbol{V} = (1/2, 1/2)^{\mathsf{t}}$ and $\mu = 5\text{e}-3$. The numerical scheme that we employ is the D_2Q_9 introduced in Section 1.5.4 with domain $\Omega = [-1/2, 1]^2$ and equilibria based on the second-order expansion of the Maxwellian [Fakhari et al., 2016]

$$m_2^{\text{eq}} = V_1 \mathsf{m}_1, \qquad m_3^{\text{eq}} = V_2 \mathsf{m}_1, \qquad m_4^{\text{eq}} = (-2\lambda^2 + 3\|\boldsymbol{V}\|_2^2)\mathsf{m}_1, \qquad m_5^{\text{eq}} = -\lambda^2 V_1 \mathsf{m}_1,$$

Table 3.9: Test (IV) for the 2D linear advection-diffusion equation taking $\overline{\ell} = 9$ and performing the computation using a mesh at level $\underline{\ell}$ as indicated.

	(2.40) with $\gamma = 0$			(2.40) with $\gamma = 1$			Lax-Wendroff (3.2)		
$\Delta\underline{\ell}$	$E_{\text{coa}}^{\underline{\ell}}$	$E_{\text{coa}}^{\overline{\ell}}$	D_{coa}	$E_{\text{coa}}^{\underline{\ell}}$	$E_{\text{coa}}^{\overline{\ell}}$	D_{coa}	$E_{\text{coa}}^{\underline{\ell}}$	$E_{\text{coa}}^{\overline{\ell}}$	D_{coa}
0	4.86e-2	4.86e-2	0.00e+0	4.86e-2	4.86e-2	0.00e+0	4.86e-2	4.86e-2	0.00e+0
1	4.61e-2	4.61e-2	2.79e-2	4.86e-2	4.86e-2	9.42e-5	4.80e-2	4.80e-2	8.20e-4
2	7.60e-2	7.58e-2	8.06e-2	4.86e-2	4.87e-2	3.89e-4	4.55e-2	4.56e-2	4.09e-3
3	1.65e-1	1.64e-1	1.75e-1	4.83e-2	4.87e-2	1.62e-3	3.66e-2	3.71e-2	1.71e-2
4	3.19e-1	3.16e-1	3.29e-1	4.64e-2	4.82e-2	7.49e-3	3.69e-2	4.01e-2	6.90e-2
5	5.47e-1	5.38e-1	5.51e-1	3.69e-2	4.99e-2	4.94e-2	2.27e-1	2.39e-1	2.82e-1
6	8.22e-1	8.16e-1	8.26e-1	4.44e-1	4.74e-1	5.14e-1	9.29e-1	1.00e+0	1.04e+0

$$m_6^{\text{eq}} = -\lambda^2 V_2 \mathsf{m}_1, \qquad m_7^{\text{eq}} = (\lambda^4 + 3\lambda^2 \|\boldsymbol{V}\|_2^2)\mathsf{m}_1, \qquad m_8^{\text{eq}} = (V_1^2 - V_2^2)\mathsf{m}_1, \qquad m_9^{\text{eq}} = V_1^2 V_2^2 \mathsf{m}_1.$$

We take $s_4 = s_5 = s_6 = s_7 = s_8 = s_9 = 1$ and

$$s_2 = s_3 = \left(\frac{1}{2} + \frac{3\mu}{\lambda \Delta x}\right)^{-1},$$

to enforce the diffusivity, along with $\lambda = 1$.

3.1.4.5.2 Results and discussion We perform the same kind of test than for the unidimensional problem in Section 3.1.4.3 , by taking $\overline{\ell}$ and using different minimum levels $\underline{\ell}$. The full results are on Table 3.9 and some plots of the solution in Figure 3.7 and Figure 3.8. We observe the following facts:

- Multiresolution scheme (2.40) for $\gamma = 0$. As one expects, the diffusion term is not correctly handled. This results in a non-negligible additional error D_{coa} in Table 3.9 and Figure 3.8 clearly shows that the packet is crushed way too rapidly. It is also interesting to notice that since the most important contribution to D_{coa} is an additional isotropic diffusion, the structure of D_{coa} (see the white contours on Figure 3.7) is essentially isotropic.
- Multiresolution scheme (2.40) for $\gamma = 1$. On the other hand, this method successfully copes with the diffusion phenomena, being able to have a negligible D_{coa} until $\Delta\underline{\ell} = 5$. Figure 3.8 shows a very good agreement with the expected solution and Figure 3.7 shows that the discrepancies from the reference scheme are essentially isotropic, since made up essentially of fourth-order terms which turn out to be isotropic, with additional rapidly oscillatory terms when $\Delta\underline{\ell}$ increases. This creates the dense amount of contours we can observe.
- Lax-Wendroff scheme (3.2). Again as expected, the method does not alter the diffusion terms but D_{coa} starts to be a dominant term earlier than for $\gamma = 1$, namely around $\Delta\underline{\ell} = 3$. This can be also understood when looking at Figure 3.8, where one clearly notices the alteration of the third order terms which induces a dispersive effect. This can also be seen on Figure 3.7, where the dispersive effect shows to be non-isotropic and linked with the propagation of the packet in space at finite velocity. Once $\Delta\underline{\ell}$ increases, we still observe, though way less intensely than for $\gamma = 1$, the development of high-frequency components of D_{coa}.

3.1.5 Conclusions

In Section 3.1, we have shown how to apply the classical analysis based on the equivalent equations introduced [Dubois, 2008] to the adaptive lattice Boltzmann schemes based on multiresolution. This has relied upon the so-called "reconstruction flattening" procedure. Therefore, we have been able to analyze the consistency of these methods with the target equations as for standard lattice Boltzmann methods and to find the maximal order of compliance of the adaptive scheme with the desired physics. In particular, our analysis has shown that the scheme based on $\gamma = 0$ is not accurate enough to handle the typical applications for which lattice Boltzmann schemes are designed, namely the simulation of models involving both transport and diffusion terms. The Lax–Wendroff scheme by [Fakhari and Lee, 2014] provides the minimal setting to utilize the most common lattice Boltzmann algorithms but it can yield unpredictable behavior of dispersive nature, which can threaten the stability of the method. The multiresolution scheme for $\gamma \geq 1$ proves to be the most reliable of the schemes we have analyzed,

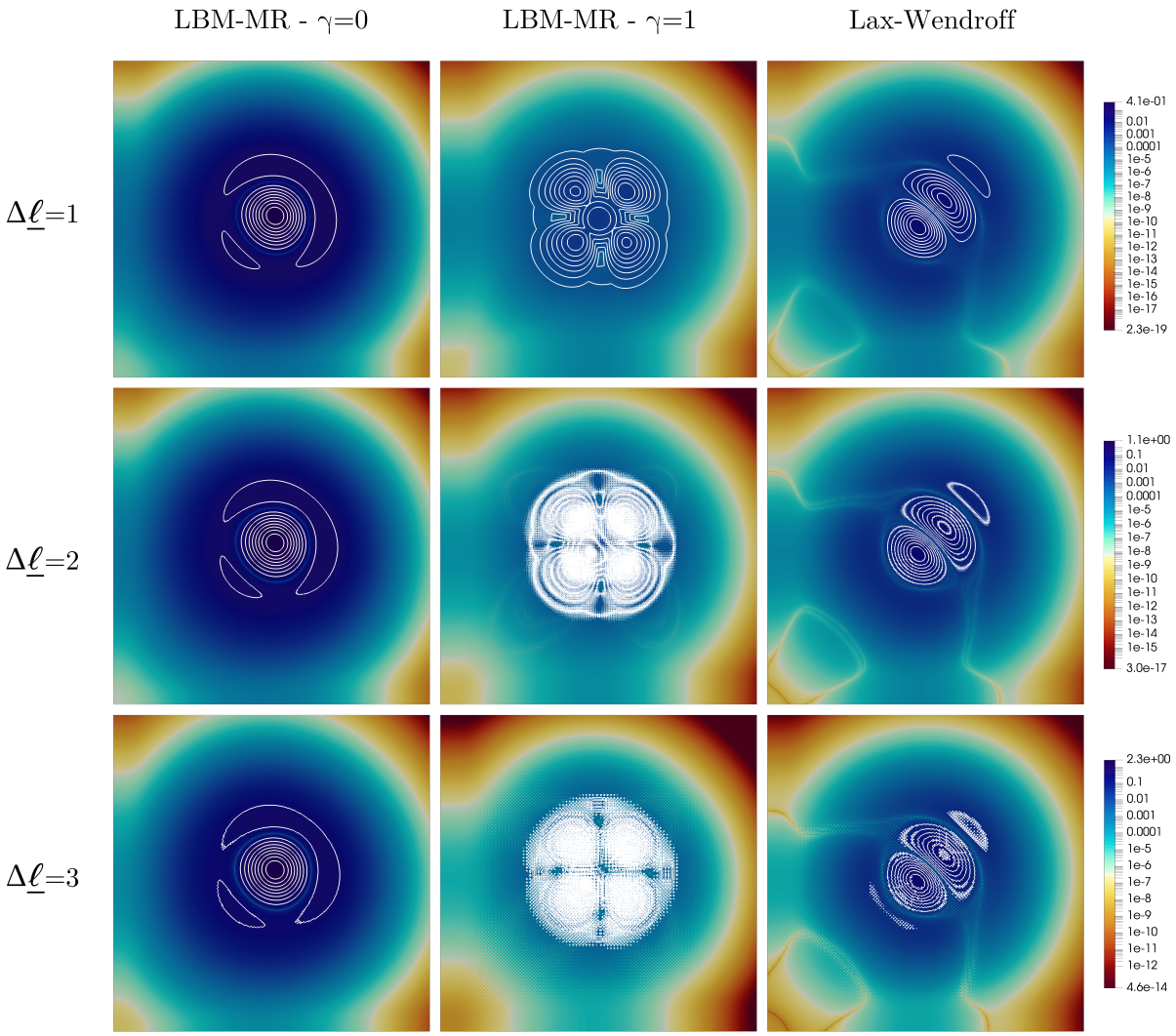


Figure 3.7: Test (IV) for the 2D linear advection-diffusion equation. Spatial patterns of D_{coa} at final time T for different $\Delta\underline{\ell}$ and stream schemes. The color scale is logarithmic. Ten contours are shown. Notice the isotropic behavior for $\gamma = 0, 1$, whereas D_{coa} is highly anisotropic for Lax-Wendroff because of the alteration of the dispersive term.

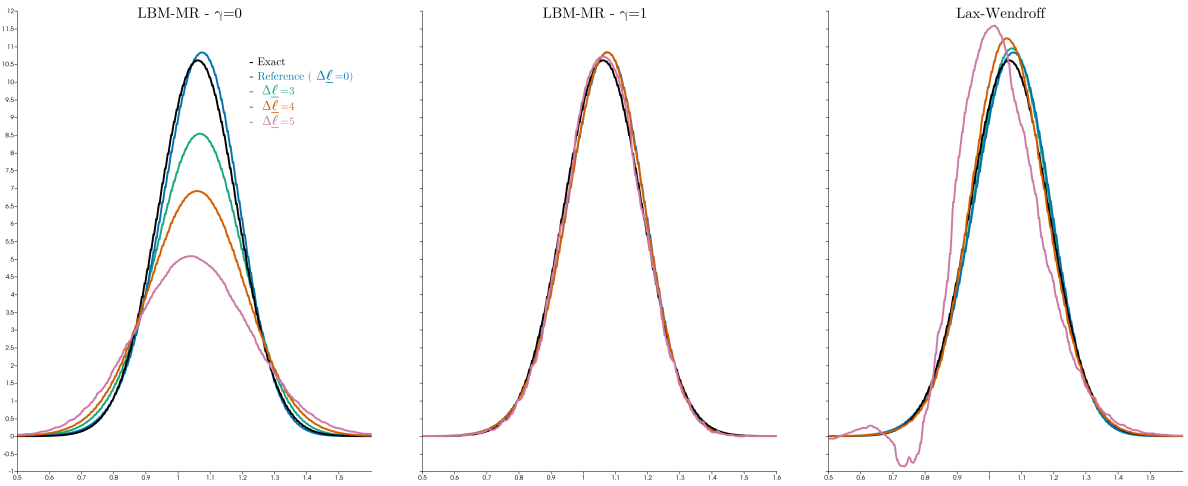


Figure 3.8: Test (IV) for the 2D linear advection-diffusion equation. Solution on a diagonal cut at final time T shown on a sector of the domain for different $\Delta\underline{\ell}$ and stream schemes. We observe the unmatched dissipation for $\gamma = 0$, excellent agreement for $\gamma = 1$ and good agreement with spurious oscillations for Lax-Wendroff.

both in terms of consistency and (quite likely) stability. The analysis is valid for locally smooth solutions. This assumption is always met on grids adapted *via* multiresolution because it guarantees a certain level of regularity of the solution at the local grid level. Our analysis has been validated using numerical simulations to solve scalar conservation laws, both linear and non-linear, in 1D and 2D, showing excellent agreement between the empirical behavior of the schemes and our asymptotic analysis. Moreover, we compared the outcome of our method against that of well-known works in literature [Fakhari and Lee, 2014, Fakhari and Lee, 2015, Fakhari et al., 2016], showing that, even if for a slightly larger computational cost, our method for $\gamma = 1$ is consistently more reliable. Finally, let us mention the fact that in Section 3.1, we have worked on uniform grids to reproduce the local environment around a given cell on general dynamically adaptive grids for smooth solutions. However, within this framework, travelling waves or shock waves leading to a lower level of regularity usually remain propagated at the finest level of mesh and thus never experience going through level jumps. However, when the mesh is fixed in advance and a level jump is present, the solution obtained with lattice Boltzmann methods is known to experience artificial reflections. The formalism proposed in Section 3.1 allows to tackle this issue as well—but requires a somewhat different setting—which we will study in Section 3.2.

3.2 Quantification of the amplitude of reflected waves at mesh jumps

The next important issue that we focus on concerns the proper treatment of acoustic waves passing through a level jump of an adapted grid. These jumps usually yield spurious effects, in particular reflected waves. We here try to quantify the amplitude of these waves.

3.2.1 1D setting

3.2.1.1 Target equations, numerical scheme and configuration

The target equation that we consider is the linear wave equation with velocity $V > 0$ on the whole real line:

$$\begin{cases} \partial_{tt} u - V^2 \partial_{xx} u = 0, & t \in [0,T], \quad x \in \mathbb{R}, \\ u(0,x) = u^\circ(x), & x \in \mathbb{R}, \\ \partial_t u(0,x) = 0, & x \in \mathbb{R}, \end{cases} \iff \begin{cases} \partial_t u + \partial_x v = 0, & t \in [0,T], \quad x \in \mathbb{R}, \\ \partial_t v + V^2 \partial_x u = 0, & t \in [0,T], \quad x \in \mathbb{R}, \\ u(0,x) = u^\circ(x), & x \in \mathbb{R}, \\ v(0,x) = 0, & x \in \mathbb{R}, \end{cases} \tag{3.19}$$

which has been recast using standard computations under the form of first order system of two conservation laws for simulating it. The simplest lattice Boltzmann scheme to handle such equation—yet yielding the difficulties of more sophisticated ones—is the D_1Q_3 scheme from Section 1.5.2 with moment matrix given by (1.5), with two conserved moments $N = 2$, which has also been employed in Section 2.8.1.4. The scheme is employed under acoustic scaling and its consistency with the target problem (3.19) is provided taking $m_3^{\mathrm{eq}}(\mathsf{m}_1) = V^2 \mathsf{m}_1$, whence $\mathsf{m}_1 \approx u$ and $\mathsf{m}_2 \approx v$. For the illustration, we shall take $T = 1.5625$, $V = 1/2$ and initial datum $u^\circ(x) = \exp(-100(x - 3/2)^2)$ for the continuous system and lattice velocity $\lambda = 1$ with $s_3 = 1.7$ for the numerical scheme.

We take a domain $\Omega = [0,3]$ which is paved using cells $C_{\ell,k}$ with $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ and $k \in [\![0, 3N_\ell[\![$. The domain is separated into two parts $\Omega_{\mathrm{left}} = [0,2]$ and $\Omega_{\mathrm{right}} = [2,3]$, so that $\Omega = \Omega_{\mathrm{left}} \cup \Omega_{\mathrm{right}}$. We consider the following numerical solutions:

- $\overline{\mathbf{m}}^{\mathrm{jump},1}$, the first conserved moment of the scheme obtained by the spatial discretization on Figure 3.9. The fixed mesh computational is obtained by meshing the left subdomain Ω_{left} finely at the maximum level $\overline{\ell}$ and the right subdomain Ω_{right} coarsely with level $\underline{\ell} \leq \overline{\ell}$. We shall vary the level gap $\Delta\underline{\ell}$. This is the $d = 1$ equivalent of the configuration presented in [Lagrava, 2012, Chapter 4].
- $\overline{\mathbf{m}}^{\mathrm{ref},1}$, the first conserved moment of the reference scheme applied on the uniform mesh at finest level $\overline{\ell}$.
- $\overline{\mathbf{m}}^{\mathrm{coarse},1}$, the first conserved moment of the scheme applied on the uniform mesh at the coarsest level $\underline{\ell}$.

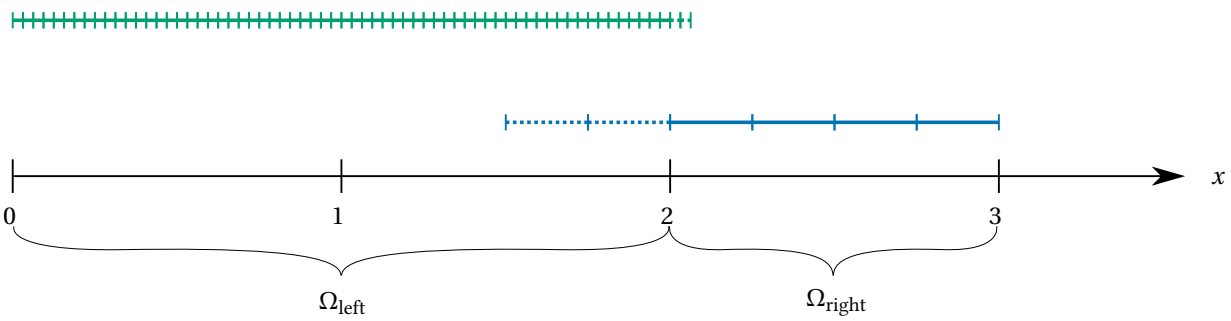


Figure 3.9: Example of domain $\Omega = \Omega_{\text{left}} \cup \Omega_{\text{right}}$ with $\Omega_{\text{left}} = [0,2]$ finely meshed (green full line) and $\Omega_{\text{right}} = [2,3]$ coarsely meshed (blue full line). Dashed lines represent ghost cells, where the solution needs to be updated to deploy the adaptive scheme.

For the purpose of normalizing, we use the norm of the exact solution $\|\overline{\boldsymbol{u}}(t)\|_{\ell^1} = \|u(t,\cdot)\|_{L^1}$, which is conserved in time, and we measure

$$
\begin{aligned}
E_{\text{ref}}(t) &:= \frac{\|\overline{\mathbf{m}}^{\text{ref},1}(t) - \overline{\boldsymbol{u}}(t)\|_{\ell^1}}{\|\overline{\boldsymbol{u}}(t)\|_{\ell^1}}, \\
E_{\text{coarse}}(t) &:= \frac{\|\hat{\overline{\mathbf{m}}}_{\overline{\ell}}^{\text{coarse},1}(t) - \overline{\boldsymbol{u}}(t)\|_{\ell^1}}{\|\overline{\boldsymbol{u}}(t)\|_{\ell^1}}, & D_{\text{coarse}}(t) &:= \frac{\|\hat{\overline{\mathbf{m}}}_{\overline{\ell}}^{\text{coarse},1}(t) - \overline{\mathbf{m}}^{\text{ref},1}(t)\|_{\ell^1}}{\|\overline{\boldsymbol{u}}(t)\|_{\ell^1}}, \\
E_{\text{jump}}(t) &:= \frac{\|\hat{\overline{\mathbf{m}}}_{\overline{\ell}}^{\text{jump},1}(t) - \overline{\boldsymbol{u}}(t)\|_{\ell^1}}{\|\overline{\boldsymbol{u}}(t)\|_{\ell^1}}, & D_{\text{jump}}(t) &:= \frac{\|\hat{\overline{\mathbf{m}}}_{\overline{\ell}}^{\text{jump},1}(t) - \overline{\mathbf{m}}^{\text{ref},1}(t)\|_{\ell^1}}{\|\overline{\boldsymbol{u}}(t)\|_{\ell^1}}, \\
D_{\text{jump-refl}}(t) &:= \frac{\|(\hat{\overline{\mathbf{m}}}_{\overline{\ell}}^{\text{jump},1}(t) - \overline{\mathbf{m}}^{\text{ref},1}(t))\mathbb{1}_{\{\overline{k}\,:\, C_{\overline{\ell},\overline{k}} \subset \Omega_{\text{left}}\}}\|_{\ell^1}}{\|\overline{\boldsymbol{u}}(t)\|_{\ell^1}}.
\end{aligned}
\tag{3.20}
$$

Here, we use the reconstruction operator—*cf.* (2.29)—to compare data at the finest level $\overline{\ell}$ also for schemes which are executed on coarsened meshes. Let us comment on these errors. E_{ref} is the error of the reference scheme. E_{coarse} is the error of the scheme on the coarse mesh and D_{coarse} is its perturbation error, that is, the difference of its solution from the reference one. E_{jump} is the error of the scheme on the mesh with a jump and D_{jump} is its perturbation error, that is, the difference of its solution from the reference one. We measure the amplitude of the reflected by $D_{\text{jump-refl}}$, which is nothing D_{jump} restricted on the left subdomain Ω_{left}, since the support of the initial datum is essentially contained in Ω_{left}, which is finely meshed.

3.2.1.2 Theoretical result on the amplitude of the reflected wave

We introduce measure of the errors and the reflected wave and prove that for $\gamma = 1$, the amplitude of the reflected wave is $O(\Delta x^4)$.

Proposition 3.2.1

Assume that the solution of (3.19) and the numerical solution of the scheme are smooth for any time between $[0,T]$, then for the adaptive scheme using (2.40) with $\gamma = 1$, we have

$$D_{\text{jump-refl}}(T) = O(\Delta x^4).$$

Proof. The idea of the proof is to use Proposition 3.1.3 and estimate the local errors in time, which add up as in the proof of the Lax equivalence theorem [Allaire, 2007]. Let us consider the following indices for the cells close to the level jump. We denote $\overline{k}_{\text{left}}$ the last cell in Ω_{left} at level $\overline{\ell}$. Then, $\overline{k}_{\text{right}}$ is the first ghost cell in Ω_{right} at level $\overline{\ell}$. Finally, $\underline{k}_{\text{right}}$ the first cell in Ω_{right} at level $\underline{\ell}$. Thanks to the smoothness assumption, we consider that the numerical solution stems from the pointwise evaluation of a smooth function defined everywhere. Therefore,

Proposition 3.1.3 provides that

$$\overline{\mathsf{f}}^{\mathrm{jump},j}_{\underline{\ell},\underline{k}_{\mathrm{right}}}(t) = f^{\mathrm{jump},j}(t, x_{\underline{\ell},\underline{k}_{\mathrm{right}}}) = f^{\mathrm{ref},j}(t, x_{\overline{\ell},\overline{k}_{\mathrm{right}}}) + O(\Delta x^4) = \overline{\mathsf{f}}^{\mathrm{ref},j}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t) + O(\Delta x^4),$$

for every time $t \in \Delta t\mathbb{N}$ and $j \in [\![1,3]\!]$. By the linearity of the collision phase, we deduce the follow post-collisional identity up to $O(\Delta x^4)$ terms

$$\overline{\mathsf{f}}^{\mathrm{jump},j,\star}_{\underline{\ell},\underline{k}_{\mathrm{right}}}(t) = \overline{\mathsf{f}}^{\mathrm{ref},j,\star}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t) + O(\Delta x^4).$$

The update of the ghost cell $C_{\overline{\ell},\overline{k}_{\mathrm{right}}}$ is done by a linear combination of values between which we also find the average on $C_{\underline{\ell},\underline{k}_{\mathrm{right}}}$, which entails

$$\overline{\mathsf{f}}^{\mathrm{jump},j,\star}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t) = \overline{\mathsf{f}}^{\mathrm{ref},j,\star}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t) + O(\Delta x^4).$$

We concentrate on $j = 2$, indexing the negatively moving velocity. Since the left subdomain is at the finest level we have

$$\begin{cases} \overline{\mathsf{f}}^{\mathrm{jump},2}_{\overline{\ell},\overline{k}_{\mathrm{left}}}(t+\Delta t) &= \overline{\mathsf{f}}^{\mathrm{jump},2,\star}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t) = \overline{\mathsf{f}}^{\mathrm{ref},2,\star}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t) + O(\Delta x^4). \\ \overline{\mathsf{f}}^{\mathrm{ref},2}_{\overline{\ell},\overline{k}_{\mathrm{left}}}(t+\Delta t) &= \overline{\mathsf{f}}^{\mathrm{ref},2,\star}_{\overline{\ell},\overline{k}_{\mathrm{right}}}(t), \end{cases}$$

therefore, computing the first conserved moment and taking the difference, we deduce that

$$\left|\overline{\mathsf{m}}^{\mathrm{jump},1}_{\overline{\ell},\overline{k}_{\mathrm{left}}}(t) - \overline{\mathsf{m}}^{\mathrm{ref},1}_{\overline{\ell},\overline{k}_{\mathrm{left}}}(t)\right| = O(\Delta x^4).$$

The CFL condition at the finest resolution $\overline{\ell}$ imposes that information—thus errors—propagate of one cell for one time step. With a constant $C > 0$ carrying the normalization in the definition of $D_{\text{jump-refl}}$, we have

$$\begin{aligned} D_{\text{jump-refl}}(T) &= C\Bigg(\overbrace{\sum_{k\in[\![0,\overline{k}_{\mathrm{left}}[\![} \Delta x\left|\overline{\mathsf{m}}^{\mathrm{jump},1}_{\overline{\ell},k}(t) - \overline{\mathsf{m}}^{\mathrm{ref},1}_{\overline{\ell},k}(t)\right|}^{\text{cells in } \Omega_{\mathrm{left}} \text{ not at the jump}} + \Delta x\left|\overline{\mathsf{m}}^{\mathrm{jump},1}_{\overline{\ell},\overline{k}_{\mathrm{left}}}(t) - \overline{\mathsf{m}}^{\mathrm{ref},1}_{\overline{\ell},\overline{k}_{\mathrm{left}}}(t)\right|\Bigg) \\ &\leq C\left(\frac{1}{C} D_{\text{jump-refl}}(T-\Delta t) + O(\Delta x^5)\right) \leq \cdots \leq \frac{T}{\Delta t} O(\Delta x^5) = \frac{\lambda T}{\Delta x} O(\Delta x^5) = O(\Delta x^4). \end{aligned}$$

□

Observe that this result would be $D_{\text{jump-refl}}(T) = O(\Delta x^2)$ for $\gamma = 0$, by virtue of Proposition 3.1.2 and $D_{\text{jump-refl}}(T) = O(\Delta x^3)$ for Lax-Wendroff (3.2), by virtue of Proposition 3.1.4.

3.2.1.3 Numerical simulation

We compare the results for (2.40) with $\gamma = 1$, the Lax-Wendroff stream (3.2): Table 3.10 and Table 3.11 provide the outcomes of the computations. Commenting on Table 3.10, we see that the reference scheme converges linearly $E_{\mathrm{ref}}(T) = O(\Delta x)$ once refining as expected from the analysis by the equivalent equations [Dubois, 2008]. The error $E_{\mathrm{coarse}}(T) \leq E_{\mathrm{ref}}(T) + D_{\mathrm{coarse}}(T) = O(\Delta x) + O(\Delta x^3) = O(\Delta x)$ converges linearly as well because the perturbation error $D_{\mathrm{coarse}}(T) = O(\Delta x^3)$ does not influence the overall convergence. The same behavior is observed for the mesh with jump, namely for $E_{\mathrm{jump}}(T)$ and $D_{\mathrm{jump}}(T)$. Very interestingly $D_{\mathrm{jump}}(T) \leq D_{\mathrm{coarse}}(T)$: counter-intuitively this is a priori not guaranteed due to the possible formation of waves reflected at the jump, even though only a part of the domain is coarsened. This gives a first indication about the fact that the reflected waves are perfectly mastered. The second indication comes from $D_{\text{jump-refl}}(T) = O(\Delta x^4)$. This means that with our method, we are able to decrease the amplitude of the reflected waves with fourth-order convergence in the space step, in accordance with Proposition 3.2.1. The supra-convergence compared to $D_{\mathrm{jump}}(T)$ comes from the fact that at each time step, the reflected wave is generated only on the cell of Ω_{right} at level $\underline{\ell}$ next to the interface, so that it eventually propagates to the left inside the fine medium without additional amplification of the error. Observe that the convergence rates worsen for large $\Delta\underline{\ell}$ and for small $\overline{\ell}$ due to the fact that we are no longer allowed to perform the Taylor expansions needed by Proposition 3.2.1, which are done at the current level of resolution ℓ. Indeed, in this case, one can no longer claim that $2^{\Delta\ell}\Delta x$ is $O(\Delta x)$. The same conclusions can be drawn for the Lax-Wendroff scheme (3.2) looking at Table 3.11, with one order less in Δx, as predicted.

Table 3.10: Results for the transition between fine and coarse mesh and (2.40) with $\gamma = 1$ as stream phase. Numerical convergence rates are reported between parenthesis.

$\bar{\ell}$	$E_{\text{ref}}(T)$	$E_{\text{coarse}}(T)$	$D_{\text{coarse}}(T)$	$E_{\text{jump}}(T)$	$D_{\text{jump}}(T)$	$D_{\text{jump-refl}}(T)$
			$\Delta\underline{\ell} = 1$			
7	7.30E-02	8.19E-02	1.30E-02	7.49E-02	2.64E-03	1.27E-05
8	3.78E-02 (0.95)	3.89E-02 (1.07)	1.86E-03 (2.81)	3.80E-02 (0.98)	3.84E-04 (2.78)	8.90E-07 (3.83)
9	1.92E-02 (0.98)	1.93E-02 (1.01)	2.28E-04 (3.03)	1.93E-02 (0.98)	5.19E-05 (2.89)	5.95E-08 (3.90)
10	9.70E-03 (0.99)	9.70E-03 (0.99)	2.49E-05 (3.20)	9.71E-03 (0.99)	6.75E-06 (2.94)	3.86E-09 (3.95)
11	4.87E-03 (0.99)	4.87E-03 (0.99)	3.67E-06 (2.76)	4.87E-03 (0.99)	8.65E-07 (2.96)	2.46E-10 (3.97)
12	2.44E-03 (1.00)	2.44E-03 (1.00)	1.08E-06 (1.77)	2.44E-03 (1.00)	1.11E-07 (2.96)	1.55E-11 (3.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	3.11E-07 (1.79)	1.22E-03 (1.00)	1.44E-08 (2.95)	9.76E-13 (3.99)
			$\Delta\underline{\ell} = 2$			
7	7.30E-02	1.61E-01	9.47E-02	9.22E-02	2.21E-02	4.84E-04
8	3.78E-02 (0.95)	5.00E-02 (1.68)	1.68E-02 (2.50)	4.04E-02 (1.19)	3.50E-03 (2.66)	3.10E-05 (3.96)
9	1.92E-02 (0.97)	2.06E-02 (1.28)	2.24E-03 (2.90)	1.96E-02 (1.05)	4.71E-04 (2.89)	2.03E-06 (3.94)
10	9.70E-03 (0.99)	9.82E-03 (1.07)	2.63E-04 (3.09)	9.74E-03 (1.01)	6.07E-05 (2.96)	1.31E-07 (3.96)
11	4.87E-03 (0.99)	4.87E-03 (1.01)	2.78E-05 (3.24)	4.88E-03 (1.00)	7.70E-06 (2.98)	8.32E-09 (3.97)
12	2.44E-03 (1.00)	2.44E-03 (1.00)	4.67E-06 (2.57)	2.44E-03 (1.00)	9.70E-07 (2.99)	5.25E-10 (3.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	1.39E-06 (1.75)	1.22E-03 (1.00)	1.22E-07 (2.99)	3.30E-11 (3.99)
			$\Delta\underline{\ell} = 3$			
7	7.30E-02	4.33E-01	3.62E-01	1.83E-01	1.12E-01	9.81E-03
8	3.78E-02 (0.95)	1.25E-01 (1.79)	9.31E-02 (1.96)	5.79E-02 (1.66)	2.26E-02 (2.31)	7.10E-04 (3.79)
9	1.92E-02 (0.97)	3.02E-02 (2.05)	1.45E-02 (2.68)	2.16E-02 (1.42)	3.12E-03 (2.86)	4.13E-05 (4.10)
10	9.70E-03 (0.99)	1.07E-02 (1.49)	1.80E-03 (3.01)	9.98E-03 (1.12)	3.97E-04 (2.98)	2.64E-06 (3.97)
11	4.87E-03 (0.99)	4.94E-03 (1.12)	1.99E-04 (3.18)	4.91E-03 (1.02)	4.96E-05 (3.00)	1.68E-07 (3.97)
12	2.44E-03 (1.00)	2.44E-03 (1.02)	2.13E-05 (3.23)	2.45E-03 (1.00)	6.19E-06 (3.00)	1.06E-08 (3.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	5.23E-06 (2.03)	1.22E-03 (1.00)	7.74E-07 (3.00)	6.66E-10 (3.99)
			$\Delta\underline{\ell} = 4$			
7	7.30E-02	9.10E-01	8.43E-01	4.00E-01	3.29E-01	6.02E-02
8	3.78E-02 (0.95)	4.24E-01 (1.10)	3.88E-01 (1.12)	1.57E-01 (1.35)	1.20E-01 (1.45)	1.43E-02 (2.07)
9	1.92E-02 (0.97)	9.69E-02 (2.13)	8.24E-02 (2.24)	3.74E-02 (2.07)	2.02E-02 (2.58)	8.21E-04 (4.13)
10	9.70E-03 (0.99)	1.78E-02 (2.44)	1.08E-02 (2.93)	1.16E-02 (1.69)	2.42E-03 (3.06)	4.42E-05 (4.22)
11	4.87E-03 (0.99)	5.47E-03 (1.71)	1.21E-03 (3.15)	5.08E-03 (1.20)	2.89E-04 (3.07)	2.85E-06 (3.95)
12	2.44E-03 (1.00)	2.45E-03 (1.16)	1.25E-04 (3.27)	2.47E-03 (1.04)	3.51E-05 (3.04)	1.83E-07 (3.97)
13	1.22E-03 (1.00)	1.21E-03 (1.02)	1.98E-05 (2.66)	1.23E-03 (1.01)	4.34E-06 (3.02)	1.15E-08 (3.99)
			$\Delta\underline{\ell} = 5$			
7	7.30E-02	1.24E+00	1.20E+00	6.46E-01	5.82E-01	1.22E-01
8	3.78E-02 (0.95)	9.30E-01 (0.41)	8.96E-01 (0.42)	3.88E-01 (0.74)	3.50E-01 (0.73)	7.35E-02 (0.73)
9	1.92E-02 (0.97)	4.51E-01 (1.04)	4.34E-01 (1.05)	1.45E-01 (1.42)	1.27E-01 (1.47)	1.86E-02 (1.98)
10	9.70E-03 (0.99)	8.05E-02 (2.49)	7.44E-02 (2.54)	2.58E-02 (2.49)	1.78E-02 (2.83)	8.87E-04 (4.39)
11	4.87E-03 (0.99)	1.06E-02 (2.93)	7.74E-03 (3.27)	6.32E-03 (2.03)	1.80E-03 (3.31)	4.31E-05 (4.36)
12	2.44E-03 (1.00)	2.68E-03 (1.98)	7.52E-04 (3.36)	2.58E-03 (1.29)	2.00E-04 (3.18)	2.84E-06 (3.93)
13	1.22E-03 (1.00)	1.18E-03 (1.18)	8.06E-05 (3.22)	1.24E-03 (1.06)	2.35E-05 (3.08)	1.86E-07 (3.93)

We also compare the solutions qualitatively including the method with local time step—which we do not describe in the manuscript—by [Rohde et al., 2006], see Figure 3.10. We see that the approach [Rohde et al., 2006], where local time-stepping is used, yields quite large reflected waves. This waves are one order of magnitude larger than for the Lax-Wendroff scheme (3.2) and two orders of magnitude larger than our approach (3.2) with $\gamma = 1$. However, the local time-stepping prevents us from applying the same theoretical study to this scheme. Mastering reflected waves at a high order of accuracy is important when our technique is extended to typical multidimensional applications. When simulating the incompressible Navier-Stokes equations *via* a quasi-incompressible D_2Q_9 scheme, *cf.* Section 2.8.2.2, spurious acoustic waves are of order $O(\Delta x^2)$, thus controlling their reflection at order $O(\Delta x^4)$ is a highly desirable feature of the scheme.

3.2.2 2D setting

3.2.2.1 Configuration and numerical scheme

The target equation we consider is quasi incompressible Navier-Stokes system (2.57), simulated with the D_2Q_9 scheme utilized in Section 2.8.2.2. The final time of the simulation is $T = 100$ in dimensionless time, the reference density $\rho_0 = 1$ and viscosity $\mu = 1.5\text{e}{-5}$. We consider the problem of the acoustic pulse presented in [Gendre et al., 2017, Astoul et al., 2021], whence the initial data are

$$\rho(0,\boldsymbol{x}) = \rho_0(1+\delta\rho_0(\boldsymbol{x})), \qquad \text{with} \qquad \delta\rho_0(\boldsymbol{x}) = \sigma\exp(-\alpha\|\boldsymbol{x}\|_2^2), \qquad \text{and} \qquad \boldsymbol{u}(0,\boldsymbol{x}) = \boldsymbol{0},$$

Table 3.11: Results for the transition between fine and coarse mesh and Lax-Wendroff (3.2) as stream phase. Numerical convergence rates are reported between parenthesis.

$\overline{\ell}$	$E_{\text{ref}}(T)$	$E_{\text{coarse}}(T)$	$D_{\text{coarse}}(T)$	$E_{\text{jump}}(T)$	$D_{\text{jump}}(T)$	$D_{\text{jump-refl}}(T)$
			$\Delta\underline{\ell} = 1$			
7	7.30E-02	1.25E-01	7.93E-02	8.17E-02	1.48E-02	1.84E-04
8	3.78E-02 (0.95)	4.62E-02 (1.43)	2.26E-02 (1.81)	3.95E-02 (1.05)	4.12E-03 (1.85)	2.37E-05 (2.95)
9	1.92E-02 (0.98)	2.05E-02 (1.17)	6.00E-03 (1.91)	1.96E-02 (1.01)	1.09E-03 (1.92)	3.08E-06 (2.95)
10	9.70E-03 (0.99)	9.91E-03 (1.05)	1.54E-03 (1.96)	9.78E-03 (1.00)	2.79E-04 (1.96)	3.95E-07 (2.96)
11	4.87E-03 (0.99)	4.91E-03 (1.01)	3.92E-04 (1.98)	4.89E-03 (1.00)	7.07E-05 (1.98)	5.00E-08 (2.98)
12	2.44E-03 (1.00)	2.45E-03 (1.00)	9.87E-05 (1.99)	2.45E-03 (1.00)	1.78E-05 (1.99)	6.30E-09 (2.99)
13	1.22E-03 (1.00)	1.22E-03 (1.00)	2.48E-05 (1.99)	1.22E-03 (1.00)	4.46E-06 (2.00)	7.90E-10 (2.99)
			$\Delta\underline{\ell} = 2$			
7	7.30E-02	4.14E-01	3.60E-01	1.36E-01	7.42E-02	3.19E-03
8	3.78E-02 (0.95)	1.28E-01 (1.70)	1.11E-01 (1.70)	5.14E-02 (1.40)	2.09E-02 (1.83)	2.47E-04 (3.69)
9	1.92E-02 (0.97)	3.70E-02 (1.79)	2.99E-02 (1.89)	2.19E-02 (1.23)	5.49E-03 (1.93)	2.99E-05 (3.04)
10	9.70E-03 (0.99)	1.25E-02 (1.57)	7.72E-03 (1.95)	1.02E-02 (1.10)	1.40E-03 (1.97)	3.79E-06 (2.98)
11	4.87E-03 (0.99)	5.29E-03 (1.24)	1.96E-03 (1.98)	4.98E-03 (1.04)	3.54E-04 (1.98)	4.79E-07 (2.98)
12	2.44E-03 (1.00)	2.51E-03 (1.08)	4.94E-04 (1.99)	2.47E-03 (1.01)	8.90E-05 (1.99)	6.03E-08 (2.99)
13	1.22E-03 (1.00)	1.23E-03 (1.02)	1.24E-04 (1.99)	1.23E-03 (1.01)	2.23E-05 (2.00)	7.56E-09 (2.99)
			$\Delta\underline{\ell} = 3$			
7	7.30E-02	1.04E+00	9.78E-01	2.98E-01	2.28E-01	3.28E-02
8	3.78E-02 (0.95)	4.57E-01 (1.19)	4.31E-01 (1.18)	1.20E-01 (1.31)	8.99E-02 (1.34)	4.53E-03 (2.85)
9	1.92E-02 (0.97)	1.31E-01 (1.81)	1.23E-01 (1.80)	3.67E-02 (1.71)	2.34E-02 (1.94)	2.50E-04 (4.18)
10	9.70E-03 (0.99)	3.46E-02 (1.92)	3.23E-02 (1.93)	1.33E-02 (1.47)	5.94E-03 (1.98)	2.95E-05 (3.08)
11	4.87E-03 (0.99)	9.62E-03 (1.85)	8.23E-03 (1.98)	5.55E-03 (1.26)	1.50E-03 (1.99)	3.73E-06 (2.98)
12	2.44E-03 (1.00)	3.18E-03 (1.60)	2.07E-03 (1.99)	2.57E-03 (1.11)	3.75E-04 (2.00)	4.72E-07 (2.98)
13	1.22E-03 (1.00)	1.33E-03 (1.26)	5.20E-04 (1.99)	1.25E-03 (1.04)	9.38E-05 (2.00)	5.93E-08 (2.99)
			$\Delta\underline{\ell} = 4$			
7	7.30E-02	1.71E+00	1.66E+00	5.07E-01	4.39E-01	9.51E-02
8	3.78E-02 (0.95)	1.15E+00 (0.57)	1.12E+00 (0.57)	2.90E-01 (0.81)	2.54E-01 (0.79)	3.96E-02 (1.26)
9	1.92E-02 (0.97)	4.84E-01 (1.25)	4.71E-01 (1.25)	1.14E-01 (1.34)	9.93E-02 (1.36)	5.53E-03 (2.84)
10	9.70E-03 (0.99)	1.32E-01 (1.87)	1.29E-01 (1.87)	3.06E-02 (1.90)	2.45E-02 (2.02)	2.38E-04 (4.54)
11	4.87E-03 (0.99)	3.40E-02 (1.96)	3.32E-02 (1.96)	9.27E-03 (1.72)	6.11E-03 (2.00)	2.71E-05 (3.13)
12	2.44E-03 (1.00)	8.80E-03 (1.95)	8.39E-03 (1.99)	3.33E-03 (1.48)	1.52E-03 (2.00)	3.43E-06 (2.98)
13	1.22E-03 (1.00)	2.43E-03 (1.86)	2.11E-03 (1.99)	1.39E-03 (1.26)	3.81E-04 (2.00)	4.39E-07 (2.97)
			$\Delta\underline{\ell} = 5$			
7	7.30E-02	1.46E+00	1.43E+00	6.73E-01	6.13E-01	1.42E-01
8	3.78E-02 (0.95)	1.81E+00 (-0.31)	1.78E+00 (-0.32)	4.97E-01 (0.44)	4.62E-01 (0.41)	1.03E-01 (0.46)
9	1.92E-02 (0.97)	1.22E+00 (0.56)	1.21E+00 (0.56)	2.86E-01 (0.80)	2.68E-01 (0.78)	4.36E-02 (1.25)
10	9.70E-03 (0.99)	4.99E-01 (1.29)	4.93E-01 (1.29)	1.12E-01 (1.36)	1.04E-01 (1.36)	6.15E-03 (2.82)
11	4.87E-03 (0.99)	1.33E-01 (1.91)	1.32E-01 (1.90)	2.79E-02 (2.00)	2.50E-02 (2.06)	2.26E-04 (4.77)
12	2.44E-03 (1.00)	3.39E-02 (1.97)	3.36E-02 (1.97)	7.60E-03 (1.88)	6.17E-03 (2.02)	2.49E-05 (3.18)
13	1.22E-03 (1.00)	8.57E-03 (1.98)	8.44E-03 (1.99)	2.32E-03 (1.71)	1.53E-03 (2.01)	3.11E-06 (3.00)

Table 3.12: Results for the acoustic pulse problem by [Gendre et al., 2017] using (2.40) with $\gamma = 1$.

$\overline{\ell}$	$E_{\text{ref}}(T)$	$E_{\text{coarse}}(T)$	$D_{\text{coarse}}(T)$	$E_{\text{jump}}(T)$	$D_{\text{jump}}(T)$	$D_{\text{jump-refl}}(T)$
6	1.48E-02	5.28E-02	4.82E-02	1.93E-02	9.19E-03	1.54E-04
7	4.52E-03	9.50E-03	6.66E-03	5.16E-03	1.49E-03	1.64E-05
8	3.07E-03	3.50E-03	8.79E-04	3.15E-03	2.77E-04	2.76E-06

with the particular choice $\sigma = 1\text{e}-3$ and $\alpha = 100 \times \log(2)$. Observe that we utilize a small viscosity, hence, in the inviscid limit $\mu \to 0$, the exact solution of (2.57) in terms of density is given by $\rho(t,\boldsymbol{x}) = \rho_0(1+\delta\rho(t,\boldsymbol{x}))$ with

$$\delta\rho(t,\boldsymbol{x}) = \frac{\sigma}{2\alpha}\int_0^{+\infty} \exp\left(\frac{-\eta^2}{4\alpha}\right)\cos\left(\frac{\lambda t\eta}{\sqrt{3}}\right)J_0(\|\boldsymbol{x}\|_2\eta)\eta\mathrm{d}\eta,$$

where J_0 is the zero order Bessel function of first kind. This solution is obtained by means of the Henkel transform. The approximation of this integral is done by using a Gauss-Laguerre approximation with one hundred points.

The configuration is taken from [Gendre et al., 2017] and based on a spatial domain $\Omega = [-1,1]^2$ with a central band in the first direction $[-12/32, 13/32] \times [-1,1]$ refined at the maximum resolution $\overline{\ell}$ and the rest of the domain at level $\underline{\ell} = \overline{\ell} - 1$, see Figure 3.11.

3.2.2.2 Numerical simulation

An example of result on the cells along the line at $x_2 = 0$ is given in Figure 3.12 for the maximum resolution of $\overline{\ell}$. We observe an excellent match between our result and the theoretical solution. For a more quantitative assessment, see Table 3.12, we repeat the test for different maximum resolutions $\overline{\ell}$ measuring different errors and differences in the L^1 norm (and not in the L^2 norm as in the literature). These are defined by (3.20) as for the test in Section 3.2.1,

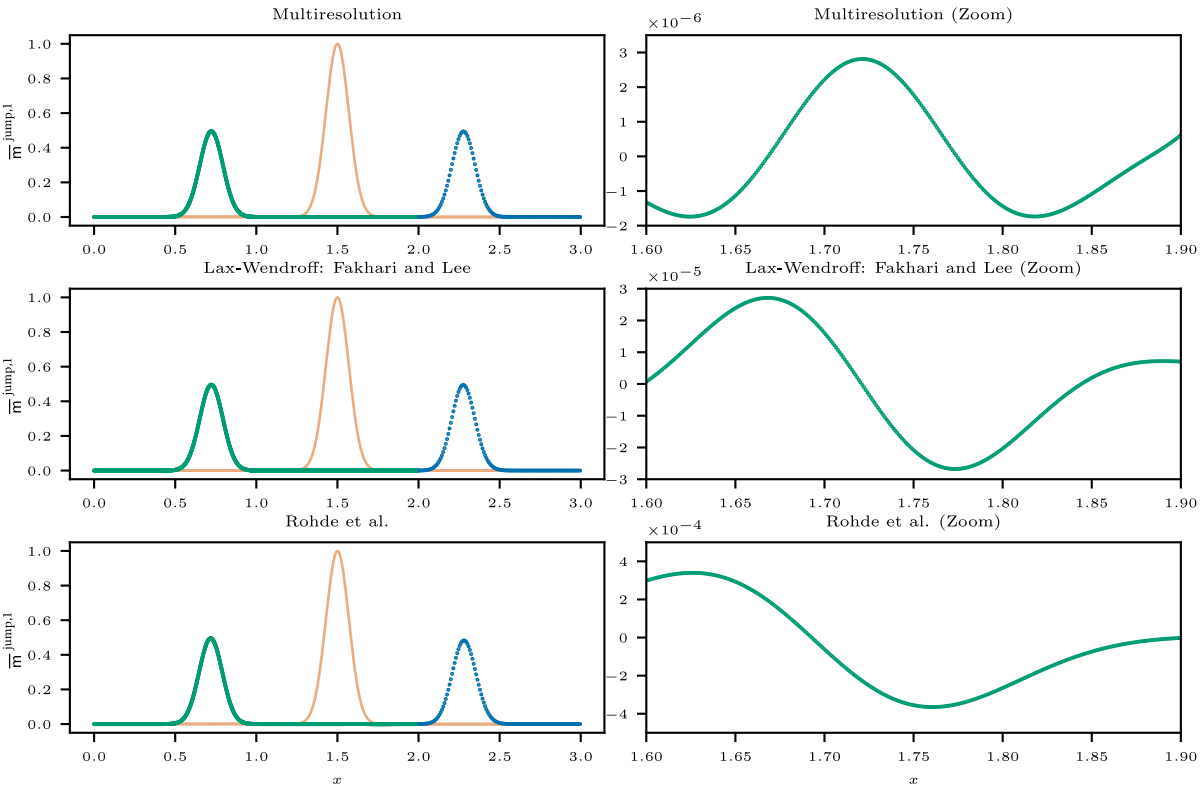


Figure 3.10: Results of the simulation on the mesh with jump (whole domain on the left, magnification on [1.6, 1.9] and on the y axis on the right). Initial solution in pale orange and solution at final time T in green (left subdomain) and blue (right subdomain). On the first row, we use our multiresolution scheme (2.40) with $\gamma = 1$. On the second row, we use the Lax-Wendroff scheme (3.2) by [Fakhari and Lee, 2014]. On the third row, the scheme with local time-stepping by [Rohde et al., 2006]. The simulation uses $\overline{\ell} = 10$ and $\Delta\underline{\ell} = 3$.

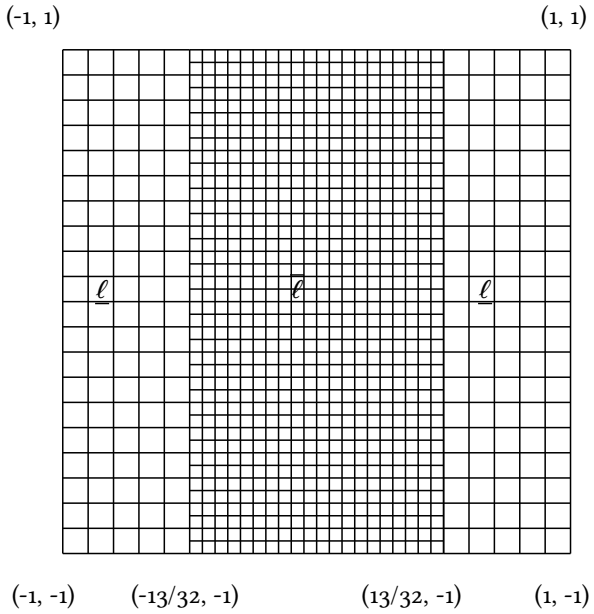


Figure 3.11: Configuration for the simulation of the acoustic pulse problem by [Gendre et al., 2017].

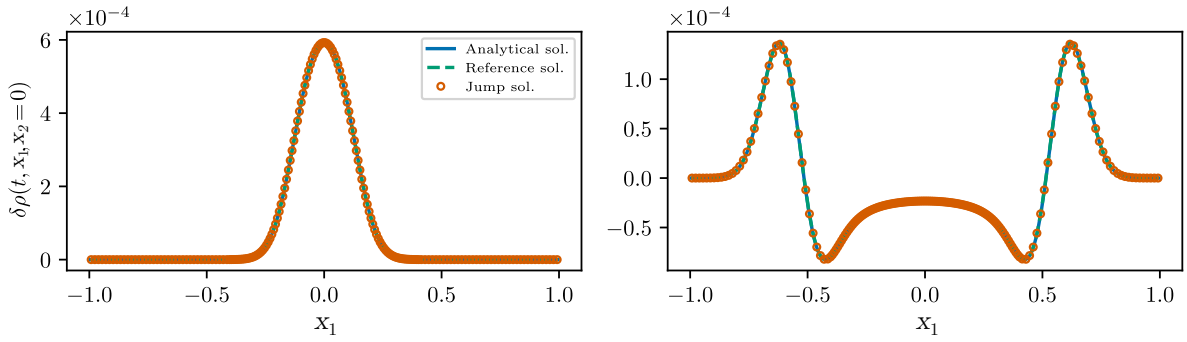


Figure 3.12: Acoustic pulse problem by [Gendre et al., 2017] using (2.40) with $\gamma = 1$ with $\overline{\ell} = 7$. We display the solution on the cut $x_2 = 0$ at the dimensionless time $t = 10$ (left) and dimensionless final time $T = 100$. We compare with the analytical solution (blue), the reference solution for the scheme at the uniform finest level $\overline{\ell}$ (green) and the adaptive solution with the mesh with level jump (orange).

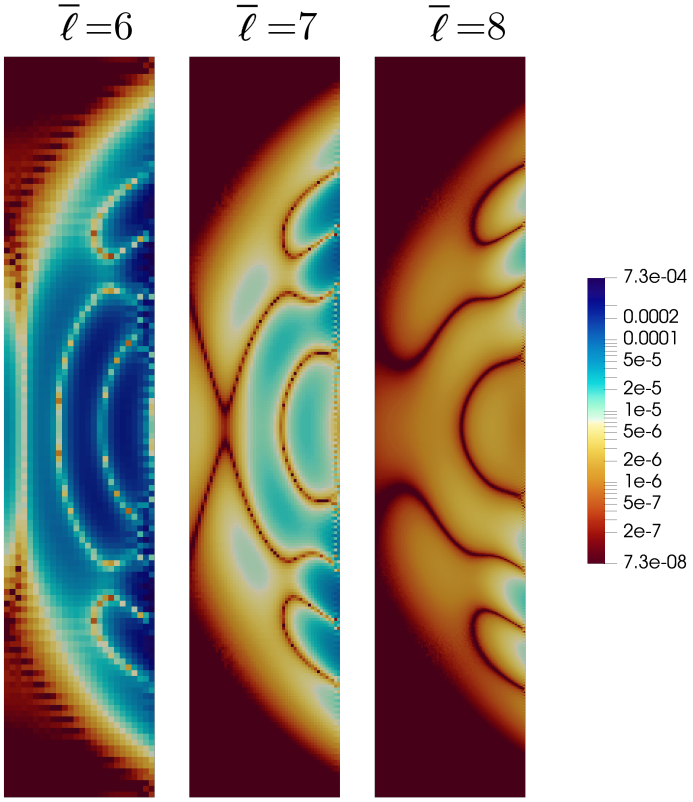


Figure 3.13: Acoustic pulse problem by [Gendre et al., 2017] using (2.40) with $\gamma = 1$ with $\overline{\ell} = 7$. We display $|\overline{\mathsf{m}}^{\mathrm{jump},1} - \overline{\mathsf{m}}^{\mathrm{ref},1}| / \|\overline{\mathbf{m}}^{\mathrm{jump},1} - \overline{\mathbf{m}}^{\mathrm{ref},1}\|_{\ell^\infty}$ at the final time T to show the reflected wave on the subdomain $[0, 13/32] \times [-1, 1]$.

by changing considering that the sum to yield $D_{\text{jump-refl}}$ is done in the central refined band $[-12/32, 13/32] \times [-1, 1]$ in order to measure the amplitude of the reflected wave. Though we do not precisely obtain a second-order slope as in [Gendre et al., 2017] (remark that we do not use the same 3D scheme, not the same norm and the discrete resolutions are not the same), we observe the convergence of all the quantities as $\overline{\ell}$ increases. Observe that for any proposed resolution, we succeed in keeping the amplitude of the reflected wave two orders of magnitude smaller than the error of the reference scheme with respect to the exact solution. In order to provide a snapshot showing the spatial structure of the reflected wave as in [Astoul et al., 2021], we show the difference between the solution for the reference scheme on the uniform lattice and that on the lattice with a jump described before, see Figure 3.13. We see that the overall discrepancy decreases with $\overline{\ell}$ as we have already seen in Table 3.12.

3.2.3 Conclusions

In Section 3.2, we have used the results on the perturbation error given in Section 3.1 to conclude that in case of a fixed mesh jump, the amplitude of the spuriously reflected waves for the adaptive lattice Boltzmann based on multiresolution with $\gamma = 1$ is of order $O(\Delta x^4)$. This fact is numerically verified and compared to the performance of other approaches available in the literature [Fakhari and Lee, 2014] and [Rohde et al., 2006], showing that our method outperforms these traditional approaches. It is worthwhile observing that the original adaptive method—*cf.* Chapter 2— was conceived to be used with dynamically adapted meshes which automatically follow waves and fronts with finer discretizations once their lack of regularity justifies the depart from a coarse uniform mesh. Thus, in this case, we even do not expect the $O(\Delta x^4)$ perturbation because fronts never cross level jumps but are precisely and successfully "chased" by the fine discretization.

3.3 Conclusions of Chapter 3

In Chapter 3, we have investigated several additional numerical properties of the strategy introduced in Chapter 2. First, we have quantified the perturbation introduced by the adaptive lattice Boltzmann scheme using an equivalent equation analysis. We have evaluated the amplitude of reflected waves originated at the mesh jumps by our method and concurrent strategies.

Chapter 4

Quantification of the perturbation error for multiresolution Finite Volume schemes

General context and motivation

Multiresolution [Mallat, 1989, Cohen et al., 2003] offers, on the one hand, an efficient way of compressing meshes while keeping the error on the information stored on the grid controlled by a threshold parameter ϵ. On the other hand, it offers a tool to devise adaptive numerical methods, such as Finite Volume, to be utilized on meshes being dynamically adapted in time, still controlling the additional error compared to the reference scheme on the uniform mesh at the finest level of available resolution [Cohen et al., 2003, Bramkamp et al., 2004, Roussel and Schneider, 2005, Dumont et al., 2013, Duarte et al., 2013, Duarte et al., 2015].

State of the art

Error control is a central feature of multiresolution. Therefore, in [Cohen et al., 2003], the perturbation introduced by the adaptive scheme based on multiresolution is quantified in terms of the threshold parameter in the case of exact local flux reconstruction. The study is extended [Hovhannisyan and Müller, 2010] to approximate flux reconstruction strategies. However, no study concerning the local truncation error, when fluxes are computed with the exact local flux reconstruction or, even worse, with the so-called "direct evaluation" [Cohen et al., 2003] is provided. The local truncation error has to be estimated using the modified equations [Warming and Hyett, 1974, Carpentier et al., 1997] of the scheme. The control of the additional error by the threshold ϵ [Cohen et al., 2003, Hovhannisyan and Müller, 2010] is interesting because it holds regardless of the smoothness of the solution, thus also when shocks and kinks are present. However, in the areas where the solution is smooth and thus can be developed in Taylor series, it is possible to analyze the additional error in more detail and with different techniques. In this zone, adaptive multiresolution adopts a coarse mesh thanks to the smoothness of the solution. A first step in this direction has been introduced in Chapter 3 in the context of lattice Boltzmann schemes. Since in these methods, the only step influenced by multiresolution is the transport phase, corresponding to upwind schemes for each discrete velocity, the analysis was essentially carried out for upwind Finite Volume schemes.

Aims and structure of Chapter 4

Here, the plan is to apply the ideas of the perturbation analysis introduced for lattice Boltzmann schemes in Section 3.1 to the traditional methods employed with multiresolution, namely the Finite Volume schemes [Cohen et al., 2003]. This allows to quantify at which order the reference scheme is perturbed as function of the prediction operator used to compute the numerical fluxes. We also aim at integrating this information in the error analysis for these methods in order to provide a more precise description of the behavior of the scheme.

To this end, Chapter 4 is structured as follows. In Section 4.1, we introduce the target problem and the basic needed formalism for Finite Volume schemes as well as the computation of their modified equations. Then, Sec-

tion 4.2 recalls—without too much details, see Chapter 2—how the numerical mesh is adapted and explains how Finite Volume schemes cope with this context. We introduce a modified equation analysis for the adaptive scheme written on the leaves of the adaptive tree structure, see Section 4.3. In Section 4.4, the modified equations are easily rewritten on the finest level of resolution, which is the ideal setting to compare errors and fosters the recovery of error estimates featuring information from the modified equations, as detailed in Section 4.5. Section 4.6 presents several numerical tests to corroborate the theoretical findings and conclusions are drawn in Section 4.7.

Contents

4.1 Target problem, discretization, Finite Volume schemes and their modified equation

4.1.1 Target problem

In Chapter 4, we are concerned with the numerical solution of the Cauchy problem associated with the linear scalar conservation law

$$\begin{cases} \partial_t u(t,x) + V\partial_x u(t,x) = 0, & (t,x) \in \mathbb{R}_+ \times \mathbb{R}, \quad (4.1) \\ u(0,x) = u^\circ(x), & x \in \mathbb{R}, \quad (4.2) \end{cases}$$

where V is the transport velocity, taken $V > 0$ without loss of generality. We limit our study to a linear framework for the sake of simplicity. Still, the study can be extended to a non-linear setting by considering Lipschitz continuous fluxes for the conservation law [Hovhannisyan and Müller, 2010]. However, one must be careful because non-linear equations can decrease the smoothness of the solution even for smooth initial data, thus putting the analysis by the modified equations out of its framework of applicability. This setting would call for the studies by [Cohen et al., 2003, Hovhannisyan and Müller, 2010] which focus on controlling the additional error by the threshold ϵ. The extension to 2D/3D problems is straightforward and done by tensorization [Bihari and Harten, 1997]. This case has been considered for lattice Boltzmann schemes—*cf.* Section 3.1.3—and yields analogous conclusions.

4.1.2 Time and space discretization

The time and space discretization follows exactly the same principles as Section 2.2. Moreover, we consider an acoustic scaling between space and time, where $\Delta x/\Delta t = \lambda > 0$ is kept fixed while $\Delta x \to 0$. Therefore, Δx is utilized as the driving discretization parameter.

4.1.3 Finite Volume schemes

We define the Finite Volume schemes we shall work with as

$$\overline{\mathrm{v}}_{\overline{\ell},\overline{k}}(t+\Delta t) = \overline{\mathrm{v}}_{\overline{\ell},\overline{k}}(t) - \frac{\Delta t}{\Delta x}\Big(\Phi(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}+1/2}(t)) - \Phi(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}-1/2}(t))\Big), \tag{4.3}$$

where we utilize the same linear numerical flux for the left and the right flux in order to enforce conservativity, thus

$$\Phi(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}-1/2}) := V\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha \overline{\mathrm{v}}_{\overline{\ell},\overline{k}+\alpha}, \qquad \Phi(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}+1/2}) := V\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha \overline{\mathrm{v}}_{\overline{\ell},\overline{k}+1+\alpha}, \tag{4.4}$$

for some flux coefficients $(\phi_\alpha)_{\alpha\in[\![\underline{\alpha},\overline{\alpha}]\!]} \subset \mathbb{R}$ determining the particular numerical scheme at hand. We shall indicate by $\mathbf{E}$ the operator associated with the reference scheme, so that $\overline{\mathbf{v}}(t+\Delta t) = \mathbf{E}\overline{\mathbf{v}}(t)$.

Remark 4.1.1. *Though we consider schemes under the form (4.3), the study that we shall develop accomodates discretizations based on the method-of-lines without difficulty, since the perturbation introduced by multiresolution uniquely pertains to the error in space, whereas the time step is global.*

Let us provide three examples of schemes that shall be used throughout the entire Chapter 4.

Example 4.1.1 (Upwind scheme)**.** *The upwind scheme is such that $\underline{\alpha} = \overline{\alpha} = -1$ and $\phi_{-1} = 1$.*

Example 4.1.2 (Lax-Wendroff scheme)**.** *The Lax-Wendroff scheme is such that $\underline{\alpha} = -1$, $\overline{\alpha} = 0$ and $\phi_{-1} = (1+V/\lambda)/2$, $\phi_0 = (1-V/\lambda)/2$.*

Example 4.1.3 (OS-3, [Daru and Tenaud, 2004])**.** *The OS-3 scheme is such that $\underline{\alpha} = -2$, $\overline{\alpha} = 0$ and $\phi_{-2} = -(1-V^2/\lambda^2)/6$, $\phi_{-1} = 1-(1-V/\lambda)/2+(1-V^2/\lambda^2)/3$ and $\phi_0 = (1-V/\lambda)/2-(1-V^2/\lambda^2)/6$.*

4.1.4 Modified equations

We can now recall how the modified equations of the reference scheme (4.3) are classically found. The total flux, ignoring the time indices, reads

$$\Phi(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}+1/2}) - \Phi(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}-1/2}) = V\sum_{\alpha=-\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha(\overline{\mathrm{v}}_{\overline{\ell},\overline{k}+\alpha+1} - \overline{\mathrm{v}}_{\overline{\ell},\overline{k}+\alpha}). \tag{4.5}$$

The analysis of the modified equation, as far as the numerical flux is concerned, is obtained by applying the scheme to a smooth function u—which is not necessarily the solution of the target PDE—evaluated at the cell centers $x_{\overline{\ell},\overline{k}}$ and then performing Taylor expansions. Applying this procedure to (4.5) provides

$$\Big(V\sum_{h=1}^{+\infty}\frac{\Delta x^h}{h!}\partial_x^h\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}}\phi_\alpha\Big((\alpha+1)^h-\alpha^h\Big)\Big)u(x_{\overline{\ell},\overline{k}}) = \Big(V\sum_{h=1}^{+\infty}\frac{\Delta x^h}{h!}\partial_x^h\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}}\phi_\alpha\Delta[\cdot^h](\alpha)\Big)u(t,x_{\overline{\ell},\overline{k}}), \tag{4.6}$$

where we have introduced the forward finite difference operator $\Delta[f](r) := f(r+1)-f(r)$, that we shall extensively employ in what follows as a shorthand. Coming back to the previous examples, we have

Example 4.1.4 (Upwind scheme)**.** *For the upwind scheme $\sum_{\alpha=\underline{\alpha}}^{\alpha=\overline{\alpha}}\phi_\alpha\Delta[\cdot^h](\alpha) = (-1)^{h+1}$ for $h \geq 1$.*

Example 4.1.5 (Lax-Wendroff scheme)**.** *For the Lax-Wendroff scheme $\sum_{\alpha=\underline{\alpha}}^{\alpha=\overline{\alpha}}\phi_\alpha\Delta[\cdot^h](\alpha) = (1+(-1)^{h+1})/2 - V/\lambda(1-(-1)^{h+1})/2$ for $h \geq 1$.*

Example 4.1.6 (OS-3). *For the OS-3 scheme* $\sum_{\alpha=\underline{\alpha}}^{\alpha=\overline{\alpha}} \phi_\alpha \Delta[\cdot^h](\alpha) = (1-V^2/\lambda^2)(2^h-1)(-1)^{h+1}/6 + (1-(1-V/\lambda)/2 + (1-V^2/\lambda^2)/3)(-1)^{h+1} + (1-V/\lambda)/2 - (1-V^2/\lambda^2)/6$.

Applying the scheme (4.3) to a smooth function of space and time and carrying out the Taylor expansions in the two variables around the center of the cell gives

$$\partial_t u(t, x_{\overline{\ell},k}) + V\Big(\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha \Delta[\cdot](\alpha)\Big)\partial_x u(t, x_{\overline{\ell},k}) = -\sum_{h=2}^{+\infty} \frac{\Delta x^{h-1}}{h!}\Big(\frac{1}{\lambda^{h-1}}\partial_t^h - V\partial_x^h \sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha \Delta[\cdot^h](\alpha)\Big) u(t, x_{\overline{\ell},k}). \tag{4.7}$$

The scheme is consistent with (4.1) upon having $\sum_{\alpha=\underline{\alpha}}^{\alpha=\overline{\alpha}} \phi_\alpha \Delta[\cdot](\alpha) = 1$, which is the case for all the schemes considered here. Re-injecting the equation truncated at previous orders into itself to eliminate the temporal derivatives on the right hand side of (4.7) yields the modified equation [Warming and Hyett, 1974, Equation (1.7)]. Practically this can be performed as in [Carpentier et al., 1997] or using the Cauchy-Kowalewski procedure [Harten et al., 1987]. We write these equations as

$$\partial_t u(t, x_{\overline{\ell},\overline{k}}) + V\partial_x u(t, x_{\overline{\ell},\overline{k}}) = \sum_{h=2}^{+\infty} \Delta x^{h-1} \sigma_h \partial_x^h u(t, x_{\overline{\ell},\overline{k}}), \tag{4.8}$$

for some coefficients $(\sigma_h)_{h\geq 1} \subset \mathbb{R}$. Then we have the order of the reference scheme, also being the order of the local truncation error, defined by $\theta := \min\{h \,:\, \sigma_h \neq 0\} \geq 1$. By the Lax theorem [Allaire, 2007], if the scheme is stable with respect to a chosen norm $\|\cdot\|$, then, for smooth solutions, the convergence rate is given by $\|\overline{\boldsymbol{u}}_{\overline{\ell}}(t) - \overline{\mathbf{v}}(t)\| \leq C_{\text{ref}} t \Delta x^\theta$, where $\overline{\boldsymbol{u}}_{\overline{\ell}}$ is the discretization by averages on the cells at finest level of resolution $\overline{\ell}$ of the exact solution to (4.1) and $C_{\text{ref}} = C_{\text{ref}}((\phi_\alpha)_\alpha, \lambda, V)$ depends on the numerical scheme at hand.

Example 4.1.7 (Upwind scheme). *For the upwind scheme,* (4.8) *reads*

$$\partial_t u + V\partial_x u = \frac{\Delta x V}{2}(1 - V/\lambda)\partial_{xx} u + O(\Delta x^2), \qquad \textit{hence} \qquad \theta = 1.$$

Example 4.1.8 (Lax-Wendroff scheme). *For the Lax-Wendroff scheme,* (4.8) *reads*

$$\partial_t u + V\partial_x u = -\frac{\Delta x^2 V}{6}(1 - V^2/\lambda^2)\partial_x^3 u + O(\Delta x^3), \qquad \textit{hence} \qquad \theta = 2.$$

Example 4.1.9 (OS-3). *For the OS-3 scheme,* (4.8) *reads*

$$\partial_t u + V\partial_x u = \frac{\Delta x^3 V}{24}(-V^3/\lambda^3 + 2V^2/\lambda^2 + V/\lambda - 2)\partial_x^4 u + O(\Delta x^4), \qquad \textit{hence} \qquad \theta = 3.$$

4.2 Adaptive Finite Volume schemes

4.2.1 Algorithm

The ingredients to perform the multiresolution transform allowing for the computation of the details as well as for the mesh thresholding are exactly the same as in Section 2.3. However, compared to Section 2.4, the order of the application of the enlargement operator $\mathcal{H}_\epsilon$ and the thresholding operator $\mathcal{T}_\epsilon$ is classically slightly different for Finite Volume. This does not create any difference with the way of proceeding of Section 2.4 except at the first time step. Following [Cohen et al., 2003, Section 3.3], given a tree structure $\Lambda(t)$ and the solution reconstructed at the finest level $\hat{\overline{\mathbf{w}}}_{\overline{\ell}}(t)$, we store the solution on the leaves of $\Lambda(t)$, which is $(\overline{w}_{\ell,k}(t))_{(\ell,k)\in S(\Lambda(t))}$. A step of the scheme works as follows.

1. **Refinement**. A new graded tree $\tilde{\Lambda}(t+\Delta t) = (\mathcal{G}\circ\mathcal{H}_\epsilon)(\Lambda(t))$ is built following [Cohen et al., 2003, Hovhannisyan and Müller, 2010] and essentially the criteria of Section 2.4.2 based on the details of $(\overline{w}_{\ell,k}(t))_{(\ell,k)\in S(\Lambda(t))}$, with procedure that might be take the size of the stencil of the Finite Volume scheme into account [Harten, 1995]. After this step, the solution at the previous time t is adapted on the new mesh encoded by $\tilde{\Lambda}(t+\Delta t)$

and actually stored as $(\overline{\mathrm{w}}_{\ell,k}(t))_{(\ell,k)\in S(\tilde{\Lambda}(t+\Delta t))}$. This reads, on the reconstructed solution

$$\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t) = A_{\tilde{\Lambda}(t+\Delta t)}\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t).$$

2. **Evolution**. The adapting scheme transforming $(\overline{\mathrm{w}}_{\ell,k}(t))_{(\ell,k)\in S(\tilde{\Lambda}(t+\Delta t))}$ into the solution at the new time $(\overline{\mathrm{w}}_{\ell,k}(t+\Delta t))_{(\ell,k)\in S(\tilde{\Lambda}(t+\Delta t))}$ reads

$$\overline{\mathrm{w}}_{\ell,k}(t+\Delta t) = \overline{\mathrm{w}}_{\ell,k}(t) - \frac{\Delta t}{\Delta x_\ell}\Big(\Phi(\hat{\hat{\overline{\mathrm{w}}}}_{\overline{\ell},2^{\Delta\ell}(k+1)+1/2}(t)) - \Phi(\hat{\hat{\overline{\mathrm{w}}}}_{\overline{\ell},2^{\Delta\ell}k-1/2}(t))\Big), \tag{4.9}$$

where the double hat operator denotes the reconstruction operator (2.29). The quantities involved in the computation of the fluxes are in general not available but need to be reconstructed from those of $S(\tilde{\Lambda}(t+\Delta t))$ (plus some ghost cells). After applying (4.9), the multiresolution transform can be used to recover the reconstruction at the finest level $\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t+\Delta t)$. We indicate $\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t+\Delta t) = \mathbf{E}_{\tilde{\Lambda}(t+\Delta t)}\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t)$, where $\mathbf{E}_{\tilde{\Lambda}(t+\Delta t)}$ averages the old solution reconstructed at the finest level on the leaves $S(\tilde{\Lambda}(t+\Delta t))$, applies (4.9) and then reconstructs it at the finest level using the multiresolution transform.

3. **Coarsening**. We take the thresholded tree $\Lambda(t+\Delta t) = \mathcal{T}_\epsilon(\tilde{\Lambda}(t+\Delta t))$ as illustrated in Section 2.3 and where the details are computed using $(\overline{\mathrm{w}}_{\ell,k}(t+\Delta t))_{(\ell,k)\in S(\tilde{\Lambda}(t+\Delta t))}$. With this, the solution is adapted on the new tree, becoming $(\overline{\mathrm{w}}_{\ell,k}(t+\Delta t))_{(\ell,k)\in S(\Lambda(t+\Delta t))}$. This reads, on the reconstructed solution

$$\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t+\Delta t) = A_{\Lambda(t+\Delta t)}\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t+\Delta t).$$

Overall, one step of the algorithm can be written as $\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t+\Delta t) = A_{\Lambda(t+\Delta t)}\mathbf{E}_{\tilde{\Lambda}(t+\Delta t)}A_{\tilde{\Lambda}(t+\Delta t)}\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t)$. Let us comment once again on the difference between $\tilde{\Lambda}(t+\Delta t)$ and $\Lambda(t+\Delta t)$. When constructing $\tilde{\Lambda}(t+\Delta t)$, the adaptive mesh is enlarged using some criteria to anticipate the possible blowups of the solution which are to expect for non-linear conservation laws. This ensures the so-called Harten heuristics (or reliability condition), *cf.* Assumptions 2.4.1, which roughly states that $S(\tilde{\Lambda}(t+\Delta t))$ is constructed such that the error estimates by ϵ are guaranteed both for the solution at time t (known when $\tilde{\Lambda}(t+\Delta t)$ is constructed) and at time $t+\Delta t$ (unknown when the mesh refinement is implemented). This rewrites as $\Lambda(t+\Delta t) \subset \tilde{\Lambda}(t+\Delta t)$. For a detailed discussion of the way of enlarging the mesh in connection with the fulfillment of the Harten heuristic, the interested reader can consult [Cohen et al., 2003].

4.2.2 Reconstruction operator and adaptive scheme

For the computations of the fluxes in the adaptive scheme (4.9), we consider a reconstruction operator (2.29) which might be constructed with a different prediction operator Definition 2.3.2 than the one to adapt the mesh. Let us say that it is generated by the recursive application of the prediction operator $\boldsymbol{P}_\triangle$ taking $\hat{\gamma}$ instead of γ until reaching information stored at the level ℓ. When $\hat{\gamma} = \gamma$, the procedure is called "exact local flux reconstruction" [Cohen et al., 2003]. When $\hat{\gamma} = 0$ but $\gamma > 0$, the approach is called "direct evaluation" [Cohen et al., 2003] or "naive evaluation' [Hovhannisyan and Müller, 2010]. The choice of $\hat{\gamma}$ influences the computational cost of the method (the larger $\hat{\gamma}$, the higher the cost) but also the quality of the numerical scheme, as we shall highlight in this Chapter.

To recover an adaptive scheme, we write the following scheme at the finest level of resolution, for any $\overline{k} \in [\![0, N_{\overline{\ell}}[\![$

$$\overline{\mathrm{w}}_{\overline{\ell},\overline{k}}(t+\Delta t) = \overline{\mathrm{w}}_{\overline{\ell},\overline{k}}(t) - \frac{\Delta t}{\Delta x}\Big(\Phi(\hat{\hat{\overline{\mathrm{w}}}}_{\overline{\ell},\overline{k}+1/2}(t)) - \Phi(\hat{\hat{\overline{\mathrm{w}}}}_{\overline{\ell},\overline{k}-1/2}(t))\Big), \quad \text{with} \quad \Phi(\hat{\hat{\overline{\mathrm{w}}}}_{\overline{\ell},\overline{k}-1/2}) := V\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}}\phi_\alpha\hat{\hat{\overline{\mathrm{w}}}}_{\overline{\ell},\overline{k}+\alpha}, \tag{4.10}$$

which is extremely similar to (4.3) and where the reconstruction of the fluxes is performed using the data on $S(\tilde{\Lambda}(t+\Delta t))$, which are nothing but the projection of $\hat{\hat{\overline{\mathbf{w}}}}_{\overline{\ell}}(t)$ over $S(\tilde{\Lambda}(t+\Delta t))$. This is what makes this scheme different from $\mathbf{E}$. We shall indicate it by $\overline{\mathbf{E}}_{\tilde{\Lambda}(t+\Delta t)}$. Observe that this is the Harten's scheme [Harten, 1995] when $\hat{\gamma} = \gamma$ and the solution at the previous time step has not undergone any thresholding, see [Cohen et al., 2003, Equation (59)].

Let now $(\ell,k)\in S(\tilde\Lambda(t+\Delta t))$. Taking the projection of (4.10) for $\overline{k}\in[\![2^{\Delta\ell}k,2^{\Delta\ell}(k+1)[\![$ on the leaf in $S(\tilde\Lambda(t+\Delta t))$ yields the multiresolution scheme (4.9), which reads

$$\overline{\mathsf{w}}_{\ell,k}(t+\Delta t)=\overline{\mathsf{w}}_{\ell,k}(t)-\frac{\Delta t}{\Delta x_\ell}\Big(\Phi(\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}(k+1)+1/2}(t))-\Phi(\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}k-1/2}(t))\Big),$$

$$\text{with}\quad \Phi(\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}k-1/2}):=V\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}}\phi_\alpha\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}k+\alpha},\quad(4.11)$$

observing that the fluxes inside the cell simplify because they sum up in a telescopic fashion. Of course, some information is loss between (4.10) and (4.11) because of the averaging procedure. Therefore, there are two different schemes we can consider for the computation of the modified equations, the first being transformed into the second by means of an average on the leaves:

- (4.10), which is better suited to integrate the modified equations into the error analysis, as done in Section 4.4.
- (4.11), being the scheme we actually deploy. Its modified equations are studied in Section 4.3.

Still, the gap between the analyses for the two schemes is easily bridged by means of a scale relation, see Lemma 4.4.1 and the modified equations for the two approaches are shown to be the same at any order.

4.3 Maximal match order between adaptive scheme on the leaves and reference scheme

The scheme given by (4.9) is practically the one which is applied to the solution defined on the leaves. It is therefore natural to start analyzing the modified equation for this scheme. However, this is only a preliminary yet useful step to perform a rigorous error analysis to be merged with the one concerning the mesh adaptation phase, because errors need to be estimated on the uniform mesh at the finest level of resolution $\overline{\ell}$. We shall deal with this point in Section 4.4.

4.3.1 Reconstruction flattening

As in Section 3.1.1.3, we observe that the action of the reconstruction operator employed in the fluxes can be "flattened" and rewritten on the local level of resolution. Consider a leaf $(\ell,k)\in S(\tilde\Lambda(t+\Delta t))$. Thanks to the linearity of the prediction operator $\boldsymbol{P}_\triangle$, we have that

$$\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}k+\delta}=\sum_\beta F^{\Delta\ell}_{\delta \bmod 2^{\Delta\ell},\beta}\overline{\mathsf{w}}_{\ell,k+\lfloor\delta/2^{\Delta\ell}\rfloor+\beta},$$

where a mod b denotes the reminder of the integer division between a and b. Remark that the cells $C_{\ell,k+\beta}$ in this formula either belong to $S(\tilde\Lambda(t+\Delta t))$ or are some ghost cells that have to be correctly updated. Observe that $k+\lfloor\delta/2^{\Delta\ell}\rfloor$ is the indices of the parent cell at level ℓ of the cell $(\overline{\ell},2^{\Delta\ell}k+\delta)$. On the other hand, $(\delta \bmod 2^{\Delta\ell})\in[\![0,2^{\Delta\ell}[\![$ determines the number of the cell $(\overline{\ell},2^{\Delta\ell}k+\delta)$ in the list of the siblings. This form of operator comes from the invariance properties in terms of scaling and shift of multiresolution. The weights $(F^{\Delta\ell}_{r,\beta})_\beta\subset\mathbb{R}$ for any $r\in[\![0,2^{\Delta\ell}[\![$ are compactly supported and the size of the support, depending on $\hat\gamma$, is given by $\max\{|\beta|\ :\ F^{\Delta\ell}_{r,\beta}\neq 0\}\le 2\hat\gamma$ for every $\Delta\ell$. One can easily see that the bound is attained for $\Delta\ell$ large enough. We obtain

$$\Phi(\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}(k+1)+1/2})-\Phi(\hat{\hat{\overline{\mathsf{w}}}}_{\overline{\ell},2^{\Delta\ell}k-1/2})=V\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}}\phi_\alpha\sum_{|\beta|\le 2\hat\gamma}F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta}(\overline{\mathsf{w}}_{\ell,k+\lfloor\alpha/2^{\Delta\ell}\rfloor+\beta+1}-\overline{\mathsf{w}}_{\ell,k+\lfloor\alpha/2^{\Delta\ell}\rfloor+\beta}).$$

4.3.2 Modified equations of the adaptive scheme on the leaves

The expansion of this flux in Taylor series reads

$$\left(V\sum_{h=1}^{+\infty}\frac{\Delta x^h}{h!}\partial_x^h\sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}}\phi_\alpha 2^{\Delta\ell(h-1)}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta}\Delta[\cdot^h](\lfloor\alpha/2^{\Delta\ell}\rfloor+\beta)\right)u(t,x_{\ell,k}), \tag{4.12}$$

and has to be compared with (4.6), the expansion of the reference scheme (4.3). Observe that for the moment the comparison is merely formal as in Section 3.1 since the smooth function u is not evaluated at the same points, because cell centers do not coincide between levels, due to the volumetric standpoint. The claim is that (in Section 3.1 we did it essentially for the upwind scheme) these two terms are equal up to order $2\hat{\gamma}+1$ regardless of the choice of flux coefficients $(\phi_\alpha)_{\alpha\in[\![\underline{\alpha},\overline{\alpha}]\!]}$, thus for any scheme written under the form (4.3).

4.3.3 Maximal match orders

Theorem 4.3.1

The modified equations of the reference Finite Volume scheme (4.3) and the one of the adaptive Finite Volume scheme (4.9) are the same up to order $2\hat{\gamma}+1$ included. In other words

$$2^{\Delta\ell(h-1)}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta}\Delta[\cdot^h](\lfloor\alpha/2^{\Delta\ell}\rfloor+\beta)=\Delta[\cdot^h](\alpha),$$

for every $\Delta\ell\ge 0$, for every $\alpha\in\mathbb{Z}$ and for every $h\in[\![1,2\hat{\gamma}+1]\!]$.

Proof. The proof proceeds by weak induction on the level difference $\Delta\ell$.

- $\Delta\ell=0$. In this case, we have $F^0_{r,\beta}=\delta_{r,0}\delta_{\beta,0}$, therefore $2^{\Delta\ell(h-1)}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta}\Delta[\cdot^h](\lfloor\alpha/2^{\Delta\ell}\rfloor+\beta)=\sum_{|\beta|\le 2\hat{\gamma}}F^0_{0,\beta}\Delta[\cdot^h](\alpha+\beta)=\Delta[\cdot^h](\alpha)$ for $\alpha\in\mathbb{Z}$ and for every $h\in[\![1,2\hat{\gamma}+1]\!]$.

- We assume that the claim holds for $\Delta\ell-1\ge 0$, that is

$$2^{(\Delta\ell-1)(h-1)}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell-1}_{\alpha \bmod 2^{\Delta\ell-1},\beta}\Delta[\cdot^h](\lfloor\alpha/2^{\Delta\ell-1}\rfloor+\beta)=\Delta[\cdot^h](\alpha), \tag{4.13}$$

  for every $\alpha\in\mathbb{Z}$ and for every $h\in[\![1,2\hat{\gamma}+1]\!]$. We now want to show that this implies the same for $\Delta\ell$. Notice that, thanks to the invariance of multiresolution by spatial shift, we can consider $\alpha\in[\![0,2^{\Delta\ell}[\![$, without loss of generality, thus the claim becomes $2^{\Delta\ell(h-1)}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell}_{\alpha,\beta}\Delta[\cdot^h](\beta)=\Delta[\cdot^h](\alpha)$ for every $\alpha\in[\![0,2^{\Delta\ell}[\![$ and for every $h\in[\![1,2\hat{\gamma}+1]\!]$. Using the prediction operator gives

$$\hat{\hat{u}}_{\ell,2^{\Delta\ell}k+\alpha}=\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell}_{\alpha,\beta}u_{\ell,k+\beta}=\begin{cases}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell-1}_{\alpha,\beta}\hat{u}_{\ell+1,2k+\beta} & \alpha\in[\![0,2^{\Delta\ell-1}[\![,\\ \sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},\beta}\hat{u}_{\ell+1,2k+\beta+1} & \alpha\in[\![2^{\Delta\ell-1},2^{\Delta\ell}[\![,\end{cases}$$

$$=\begin{cases}\sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell-1}_{\alpha,\beta}\Big(u_{\ell,k+\lfloor\beta/2\rfloor}+(-1)^\beta\sum_{\delta=1}^{\hat{\gamma}}\psi_\delta(u_{\ell,k+\lfloor\beta/2\rfloor+\delta}-u_{\ell,k+\lfloor\beta/2\rfloor-\delta})\Big), & \alpha\in[\![0,2^{\Delta\ell-1}[\![,\\ \sum_{|\beta|\le 2\hat{\gamma}}F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},\beta}\Big(u_{\ell,k+\lfloor(\beta+1)/2\rfloor}+(-1)^{\beta+1}\sum_{\delta=1}^{\hat{\gamma}}\psi_\delta(u_{\ell,k+\lfloor(\beta+1)/2\rfloor+\delta}-u_{\ell,k+\lfloor(\beta+1)/2\rfloor-\delta})\Big), & \alpha\in[\![2^{\Delta\ell-1},2^{\Delta\ell}[\![,\end{cases}$$

  where the weights are given in Table 2.1. Let us treat the first case, when $\alpha\in[\![0,2^{\Delta\ell-1}[\![$. Comparing term by term we obtain the recurrence relation for the coefficients $F^{\Delta\ell}_{\alpha,\beta}$. Since the matrices we consider shall be of odd dimension, we allow to consider relative indices with respect to the central column/row. The relation

reads

$$\begin{bmatrix} F^{\Delta\ell}_{\alpha,-2\hat\gamma} \\ \vdots \\ F^{\Delta\ell}_{\alpha,-1} \\ F^{\Delta\ell}_{\alpha,0} \\ F^{\Delta\ell}_{\alpha,1} \\ \vdots \\ F^{\Delta\ell}_{\alpha,2\hat\gamma} \end{bmatrix} = \underbrace{(\boldsymbol{\psi}|\overline{\boldsymbol{\psi}}|\boldsymbol{L}\boldsymbol{\psi}|\boldsymbol{L}\overline{\boldsymbol{\psi}}|\cdots|\boldsymbol{L}^{2\hat\gamma-1}\boldsymbol{\psi}|\boldsymbol{L}^{2\hat\gamma-1}\overline{\boldsymbol{\psi}}|\boldsymbol{L}^{2\hat\gamma}\boldsymbol{\psi})}_{=:\boldsymbol{P}} \begin{bmatrix} F^{\Delta\ell-1}_{\alpha,-2\hat\gamma} \\ \vdots \\ F^{\Delta\ell-1}_{\alpha,-1} \\ F^{\Delta\ell-1}_{\alpha,0} \\ F^{\Delta\ell-1}_{\alpha,1} \\ \vdots \\ F^{\Delta\ell-1}_{\alpha,2\hat\gamma} \end{bmatrix},$$

where $\boldsymbol{\psi} = (-\psi_{\hat\gamma},\dots,-\psi_1,1,\psi_1,\dots,\psi_{\hat\gamma},0,\dots,0)^{\mathsf{t}} \in \mathbb{R}^{2\hat\gamma+1}$ and $\overline{\boldsymbol{\psi}} = (\psi_{\hat\gamma},\dots,\psi_1,1,-\psi_1,\dots,-\psi_{\hat\gamma},0,\dots,0)^{\mathsf{t}} \in \mathbb{R}^{2\hat\gamma+1}$ with the lower shift matrix $\boldsymbol{L}$ which is such that $L_{ij} = \delta_{i,j+1}$. For example, we have

$$\hat\gamma = 1, \qquad \boldsymbol{P} = \begin{bmatrix} 1/8 & -1/8 & 0 & 0 & 0 \\ 1 & 1 & 1/8 & -1/8 & 0 \\ -1/8 & 1/8 & 1 & 1 & 1/8 \\ 0 & 0 & -1/8 & 1/8 & 1 \\ 0 & 0 & 0 & 0 & -1/8 \end{bmatrix},$$

or

$$\hat\gamma = 2, \qquad \boldsymbol{P} = \begin{bmatrix} -3/128 & 3/128 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 11/64 & -11/64 & -3/128 & 3/128 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 11/64 & -11/64 & -3/128 & 3/128 & 0 & 0 & 0 \\ -11/64 & 11/64 & 1 & 1 & 11/64 & -11/64 & -3/128 & 3/128 & 0 \\ 3/128 & -3/128 & -11/64 & 11/64 & 1 & 1 & 11/64 & -11/64 & -3/128 \\ 0 & 0 & 3/128 & -3/128 & -11/64 & 11/64 & 1 & 1 & 11/64 \\ 0 & 0 & 0 & 0 & 3/128 & -3/128 & -11/64 & 11/64 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3/128 & -3/128 & -11/64 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3/128 \end{bmatrix}.$$

Hence we have

$$\begin{aligned} 2^{\Delta\ell(h-1)} \sum_{|\beta|\le 2\hat\gamma} F^{\Delta\ell}_{\alpha,\beta} \Delta[\cdot^h](\beta) &= 2^{\Delta\ell(h-1)} \sum_{|\beta|\le 2\hat\gamma} \sum_{|r|\le 2\hat\gamma} P_{\beta r} F^{\Delta\ell-1}_{\alpha,r} \Delta[\cdot^h](\beta) \\ &= 2^{\Delta\ell(h-1)} \sum_{|r|\le 2\hat\gamma} F^{\Delta\ell-1}_{\alpha,r} \sum_{|\beta|\le 2\hat\gamma} P_{\beta r} \Delta[\cdot^h](\beta). \end{aligned}$$

We observe that $\Delta[\cdot^h](\beta) = (\beta+1)^h - \beta^h = h\int_\beta^{\beta+1} x^{h-1}\mathrm{d}x$, whence

$$\begin{aligned} &2^{\Delta\ell(h-1)} \sum_{|\beta|\le 2\hat\gamma} F^{\Delta\ell}_{\alpha,\beta} \Delta[\cdot^h](\beta) = 2^{\Delta\ell(h-1)} h \sum_{|r|\le 2\hat\gamma} F^{\Delta\ell-1}_{\alpha,r} \sum_{|\beta|\le 2\hat\gamma} P_{\beta r} \int_\beta^{\beta+1} x^{h-1}\mathrm{d}x \\ &= 2^{(\Delta\ell-1)(h-1)} h \sum_{|r|\le 2\hat\gamma} F^{\Delta\ell-1}_{\alpha,r} \int_r^{r+1} x^{h-1}\mathrm{d}x = 2^{(\Delta\ell-1)(h-1)} \sum_{|r|\le 2\hat\gamma} F^{\Delta\ell-1}_{\alpha,r} \Delta[\cdot^h](r) = \Delta[\cdot^h](\alpha), \end{aligned}$$

where the second inequality comes from the special relation between the columns of $\boldsymbol{P}$ and the prediction operator and from the accuracy of the prediction operator, see Proposition 2.3.1, plus a change of variable in the integral. The last equality comes from the induction hypothesis (4.13).

Concerning the case where $\alpha \in [\![2^{\Delta\ell-1}, 2^{\Delta\ell}[\![$, again comparing term by term we obtain

$$\begin{aligned} &\left(F^{\Delta\ell}_{\alpha,-2\hat\gamma},\cdots,F^{\Delta\ell}_{\alpha,-1},F^{\Delta\ell}_{\alpha,0},F^{\Delta\ell}_{\alpha,1},\cdots,F^{\Delta\ell}_{\alpha,2\hat\gamma}\right)^{\mathsf{t}} \\ &\qquad = \boldsymbol{P}^{\mathrm{at}} \left(F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},-2\hat\gamma},\cdots,F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},-1},F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},0},F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},1},\cdots,F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},2\hat\gamma}\right)^{\mathsf{t}}, \end{aligned}$$

where $\boldsymbol{P}^{\mathrm{at}} = (\overline{\boldsymbol{\psi}}|\boldsymbol{L}\boldsymbol{\psi}|\boldsymbol{L}\overline{\boldsymbol{\psi}}|\cdots|\boldsymbol{L}^{2\hat\gamma}\boldsymbol{\psi}|\boldsymbol{L}^{2\hat\gamma}\overline{\boldsymbol{\psi}})$ is also the transpose of $\boldsymbol{P}$ along the anti-diagonal. Analogous

computations yield

$$2^{\Delta\ell(h-1)}\sum_{|\beta|\le 2\hat\gamma} F^{\Delta\ell}_{\alpha,\beta}\Delta[\cdot^h](\beta) = 2^{(\Delta\ell-1)(h-1)}\sum_{|r|\le 2\hat\gamma} F^{\Delta\ell-1}_{\alpha-2^{\Delta\ell-1},r}\Delta[\cdot^h](r+1) = \Delta[\cdot^h](\alpha),$$

again using the induction hypothesis (4.13). As a matter of fact, by invariance, we could have carried out the proof only for $\alpha \in [\![0, 2^{\Delta\ell-1}[\![$.

This achieves the proof. □

This result establishes at which order the modified equations of the reference scheme are perturbed by the introduction of the adaptive scheme. However, it does not characterize the terms in the modified equations of (4.9) above order $2\hat\gamma+1$ in Δx. For accessing these contributions (in practice, we have to compute the powers of $\boldsymbol{P}$ and $\boldsymbol{P}^{\mathrm{at}}$), symbolic computations are necessary to find them as function of $\Delta\ell$. We provide them for the three examples, which are obtained using symbolic computation relying on `sympy`.

Example 4.3.1 (Upwind scheme). *We obtain, for $\Delta\ell \in \mathbb{N}$*

$$\partial_t u + V\partial_x u = \frac{\Delta x V}{2}\left((2^{\Delta\ell} - \frac{V}{\lambda}\right)\partial_{xx}u + O(\Delta x^2), \qquad \text{for} \quad \hat\gamma = 0,$$

$$\partial_t u + V\partial_x u = \frac{\Delta x V}{2}\left(1 - \frac{V}{\lambda}\right)\partial_{xx}u - \frac{\Delta x^2 V}{6}\left(1 - \frac{V^2}{\lambda^2}\right)\partial^3_x u + \frac{\Delta x^3 V}{24}\left(-3\Delta\ell 2^{2\Delta\ell} + 2^{2\Delta\ell} - \frac{V^3}{\lambda^3}\right)\partial^4_x u + O(\Delta x^4), \qquad \text{for} \quad \hat\gamma = 1.$$

Example 4.3.2 (Lax-Wendroff scheme). *We obtain, for $\Delta\ell \in \mathbb{N}$*

$$\partial_t u + V\partial_x u = \frac{\Delta x V^2}{2\lambda}(2^{\Delta\ell} - 1)\partial_{xx}u + O(\Delta x^2), \qquad \text{for} \quad \hat\gamma = 0,$$

$$\partial_t u + V\partial_x u = -\frac{\Delta x^2 V}{6}\left(1 - \frac{V^2}{\lambda^2}\right)\partial^3_x u + \frac{\Delta x^3 V^2}{24\lambda}\left(-3\Delta\ell 2^{2\Delta\ell} + 2^{2\Delta\ell} - \frac{V^2}{\lambda^2}\right)\partial^4_x u + O(\Delta x^4), \qquad \text{for} \quad \hat\gamma = 1.$$

Example 4.3.3 (OS-3 scheme). *We obtain, for $\Delta\ell \in \mathbb{N}^*$*

$$\partial_t u + V\partial_x u = \frac{\Delta x V}{6}\left(-2^{\Delta\ell}\frac{V^2}{\lambda^2} + 3\times 2^{\Delta\ell}\frac{V}{\lambda} + 2^{\Delta\ell} - 3\frac{V}{\lambda}\right)\partial_{xx}u + O(\Delta x^2), \quad \text{for} \quad \hat\gamma = 0,$$

$$\partial_t u + V\partial_x u = \frac{\Delta x^3 V}{24}\left(-3\Delta\ell 2^{2\Delta\ell}\frac{V}{\lambda} + 2^{2\Delta\ell}\frac{V}{\lambda} + 2\times 2^{2\Delta\ell}\frac{V^2}{\lambda^2} - 2\times 2^{2\Delta\ell} - \frac{V^3}{\lambda^3}\right)\partial^4_x u + O(\Delta x^4), \quad \text{for} \quad \hat\gamma = 1.$$

The fact that the first equation is not the modified equation of the reference scheme when $\Delta\ell = 0$ is perfectly fine since the reference scheme visits two neighbors in the upwind direction, whereas the multiresolution scheme for $\hat\gamma = 0$ visits only one neighbor in the upwind direction when $\Delta\ell > 0$.

Remark 4.3.1 (Spectrum of $\boldsymbol{P}$). *In the case $\hat\gamma = 1$, the matrix $\boldsymbol{P}$ is not diagonalizable (the algebraic multiplicity of the eigenvalue $1/2$ is two) but can only be decomposed under the Jordan normal form. The eigenvalues are all real. This creates the linear terms in $\Delta\ell$ that we see in the previous modified equations. If we take $\hat\gamma = 2$, one can check that expressions for the coefficients do not contain linear terms in $\Delta\ell$, because the matrix $\boldsymbol{P}$ is diagonalizable with real eigenvalues. For $\hat\gamma = 3$, the matrix $\boldsymbol{P}$ is diagonalizable but has also complex eigenvalues. Generally speaking, there seems to be no precise regular pattern concerning the spectrum of $\boldsymbol{P}$ as $\hat\gamma$ increases and thus the possibility of easily compute its powers.*

4.4 Maximal match order between adaptive scheme at the finest level and reference scheme

As previously pointed out, though one practically employs (4.9) to evolve the solution, the error analysis must be performed on a uniform mesh at the finest level $\overline{\ell}$ because the adaptive mesh moves with time. We now adapt the

result of Theorem 4.3.1—which concerns (4.9)—to (4.10) in order to perform the analysis at the finest level.

4.4.1 Modified equations of the adaptive scheme at the finest level

Let $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ and $k \in [\![0, N_\ell[\![$. We use the projection operator $\Delta\ell$ times in the reconstruction operator to yield the datum used in the fluxes, *i.e.* the one on the leaves $S(\tilde{\Lambda}(t+\Delta t))$.

$$\hat{\hat{\mathsf{w}}}_{\overline{\ell},2^{\Delta\ell}k+\delta} = \sum_{|\beta|\le 2\hat{\gamma}} F^{\Delta\ell}_{\delta \bmod 2^{\Delta\ell},\beta}\, \overline{\mathsf{w}}_{\ell,k+\lfloor \delta/2^{\Delta\ell}\rfloor+\beta} = 2^{-\Delta\ell} \sum_{|\beta|\le 2\hat{\gamma}} F^{\Delta\ell}_{\delta \bmod 2^{\Delta\ell},\beta} \sum_{r=0}^{2^{\Delta\ell}-1} \overline{\mathsf{w}}_{\overline{\ell},2^{\Delta\ell}k+2^{\Delta\ell}(\lfloor \delta/2^{\Delta\ell}\rfloor+\beta)+r},$$

for $\delta \in \mathbb{Z}$. With a change of indices, this reads, for any $\overline{k} \in [\![0, N_{\overline{\ell}}[\![$

$$\hat{\hat{\mathsf{w}}}_{\overline{\ell},\overline{k}+\delta} = 2^{-\Delta\ell} \sum_{|\beta|\le 2\hat{\gamma}} F^{\Delta\ell}_{\delta \bmod 2^{\Delta\ell},\beta} \sum_{r=0}^{2^{\Delta\ell}-1} \overline{\mathsf{w}}_{\overline{\ell},\overline{k}+2^{\Delta\ell}(\lfloor \delta/2^{\Delta\ell}\rfloor+\beta)+r}.$$

We now consider (4.10), where the fluxes are modified by the reconstruction and computed solely using data on $S(\tilde{\Lambda}(t+\Delta t))$. For this scheme, considering that the cells of $S(\tilde{\Lambda}(t+\Delta t))$ underneath $(\overline{\ell}, \overline{k})$ are at level ℓ, we obtain

$$\begin{aligned}\Phi(\hat{\hat{\mathsf{w}}}^n_{\overline{\ell},\overline{k}+1/2}) - \Phi(\hat{\hat{\mathsf{w}}}^n_{\overline{\ell},\overline{k}-1/2}) \\ = \frac{V}{2^{\Delta\ell}} \sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha \sum_{|\beta|\le 2\hat{\gamma}} F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta} \sum_{r=0}^{2^{\Delta\ell}-1} \left(\overline{\mathsf{w}}_{\overline{\ell},\overline{k}+2^{\Delta\ell}(\lfloor \alpha/2^{\Delta\ell}\rfloor+\beta)+r+1} - \overline{\mathsf{w}}_{\overline{\ell},\overline{k}+2^{\Delta\ell}(\lfloor \alpha/2^{\Delta\ell}\rfloor+\beta)+r} \right),\end{aligned}$$

therefore its expansion in Taylor series to be compared with (4.12) and eventually with (4.6) reads

$$\left(V \sum_{h=1}^{+\infty} \frac{\Delta x^h}{h!} \partial_x^h \sum_{\alpha=\underline{\alpha}}^{\overline{\alpha}} \phi_\alpha 2^{-\Delta\ell} \sum_{|\beta|\le 2\hat{\gamma}} F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta} \sum_{r=0}^{2^{\Delta\ell}-1} \Delta[\cdot^h](2^{\Delta\ell}(\lfloor \alpha/2^{\Delta\ell}\rfloor+\beta)+r) \right) u(t, x_{\overline{\ell},\overline{k}}). \tag{4.14}$$

Notice that now the centers of the cells now coincide between (4.14) and (4.6), because we are comparing everything at the finest level $\overline{\ell}$ of resolution.

4.4.2 Modified equation at the finest level *vs* on the leaves

We prove, using a scale relation, that the modified equations for the adaptive scheme at the finest level, *i.e.* (4.14), and the one on the leaves (4.12) are the same at any order.

Lemma 4.4.1

The modified equations of the adaptive Finite Volume scheme (4.9) and the Harten-like Finite Volume scheme (4.10) coincide at any order, namely

$$\sum_{r=0}^{2^{\Delta\ell}-1} \Delta[\cdot^h](2^{\Delta\ell}(\lfloor \alpha/2^{\Delta\ell}\rfloor+\beta)+r) = 2^{h\Delta\ell} \Delta[\cdot^h](\lfloor \alpha/2^{\Delta\ell}\rfloor+\beta),$$

for $\Delta\ell \in \mathbb{N}$, $h \in \mathbb{N}$, $\alpha \in \mathbb{Z}$ and $\beta \in \mathbb{Z}$.

Proof. The thesis can be restated as

$$\sum_{r=0}^{2^{\Delta\ell}-1} \Delta[\cdot^h](2^{\Delta\ell}p+r) = 2^{h\Delta\ell} \Delta[\cdot^h](p), \tag{4.15}$$

for $\Delta\ell \in \mathbb{N}$, $h \in \mathbb{N}$ and $p \in \mathbb{Z}$, which is true by telescopic sum

$$\sum_{r=0}^{2^{\Delta\ell}-1} \Delta[\cdot^h](2^{\Delta\ell}p+r) = \sum_{r=0}^{2^{\Delta\ell}-1} ((2^{\Delta\ell}p+r+1)^h - (2^{\Delta\ell}p+r)^h) = ((2^{\Delta\ell}p+2^{\Delta\ell})^h - (2^{\Delta\ell}p)^h) = 2^{h\Delta\ell}\Delta[\cdot^h](p).$$

□

4.4.3 Maximal match orders

We deduce, as a Corollary of Theorem 4.3.1, that the modified equations are the same (match) until order $2\hat{\gamma}+1$.

Corollary 4.4.1

The modified equation of the reference Finite Volume scheme (4.3) and the one of the Harten-like Finite Volume scheme (4.10) are the same up to order $2\hat{\gamma}+1$ included. In other words

$$2^{-\Delta\ell} \sum_{|\beta|\leq 2\hat{\gamma}} F^{\Delta\ell}_{\alpha \bmod 2^{\Delta\ell},\beta} \sum_{r=0}^{2^{\Delta\ell}-1} \Delta[\cdot^h](2^{\Delta\ell}(\lfloor \alpha/2^{\Delta\ell} \rfloor + \beta) + r) = \Delta[\cdot^h](\alpha),$$

for every $\Delta\ell \geq 0$, for every $\alpha \in \mathbb{Z}$ and for every $h \in [\![1, 2\hat{\gamma}+1]\!]$.

Proof. This immediately follows from Theorem 4.3.1 and Lemma 4.4.1 applied until order $2\hat{\gamma}+1$. □

4.5 Error estimate

We prove an error estimate that implies the choice $\hat{\gamma}$. We can now use Corollary 4.4.1 to control the perturbation error introduced by the adaptive method. This is summarized in the following result.

Theorem 4.5.1

Assume that

- The reference scheme satisfies the restricted stability condition $|||\mathbf{E}||| \leq 1$, where $|||\cdot|||$ is the induced norm by the norm $\|\cdot\|$ at hand.
- The Harten-like scheme satisfies the restricted stability condition $|||\overline{\mathbf{E}}_\Lambda||| \leq 1$ for any Λ.

Then, for smooth solution, in the limit $\Delta x \to 0$ (*i.e.* $\overline{\ell} \to +\infty$) and for $\Delta\underline{\ell} = \overline{\ell} - \underline{\ell}$ kept fixed, we have the error estimate

$$\|\overline{\mathbf{v}}(t) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t)\| \leq \left(C_{\mathrm{tr}} \Delta x^{2\hat{\gamma}+1} + C_{\mathrm{MR}} \frac{\lambda}{\Delta x} \epsilon \right) t,$$

where $C_{\mathrm{tr}} = C_{\mathrm{tr}}(\overline{\ell} - \underline{\ell}, (\phi_\alpha)_\alpha, \lambda, \hat{\gamma}, V)$ and $C_{\mathrm{MR}} = C_{\mathrm{MR}}(\overline{\ell} - \underline{\ell}, (\phi_\alpha)_\alpha, \lambda, \hat{\gamma}, \gamma, V)$. Hence also

$$\|\overline{\boldsymbol{u}}_{\overline{\ell}}(t) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t)\| \leq \left(C_{\mathrm{ref}} \Delta x^{\theta} + C_{\mathrm{tr}} \Delta x^{2\hat{\gamma}+1} + C_{\mathrm{MR}} \frac{\lambda}{\Delta x} \epsilon \right) t.$$

Let us start by discussing the assumption that we have placed in the statement of Theorem 4.5.1:

- The restricted stability condition $|||\mathbf{E}||| \leq 1$ could be replaced by a milder condition $|||\mathbf{E}||| \leq 1 + C\Delta t$ for some constant $C \geq 0$, see [Cohen et al., 2003, Equation (69)] and [Hovhannisyan and Müller, 2010, Equation (A2)]. This would not change the result. The technical assumption $|||\overline{\mathbf{E}}_\Lambda||| \leq 1$ is harder to relax and also difficult to check in practice.
- The fact of considering smooth solutions comes from the fact that we want to apply the analysis of the modified equations to obtain the convergence rates, in the spirit of the Lax theorem [Allaire, 2007]. For the same reason, we take $\Delta x \to 0$ (or $\overline{\ell} \to +\infty$).
- The distance between maximum and minimum level $\Delta\underline{\ell} = \overline{\ell} - \underline{\ell}$ has to be fixed, because otherwise the constant C_{tr} potentially explodes and dominates $\Delta x^{2\hat{\gamma}+1}$ when $\Delta x \to 0$. This would prevent us from comparing orders. Moreover, this is also reasonable from the standpoint of actual computations, where we refine the mesh to achieve convergence (or nearly so) keeping the number of different available grid levels fixed. Still,

we shall also perform numerical demonstration without fixing $\Delta\underline{\ell} = \overline{\ell} - \underline{\ell}$ to show that the modified equations that we have previously developed provide important information on the behavior of $\|\overline{\mathbf{v}}(t) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t)\|$.

We continue by commenting the thesis of Theorem 4.5.1:

- The error estimate contains three contributions: the discretization error of the reference scheme, the perturbation error between the reference and the adaptive scheme, and the thresholding error coming from the multiresolution used to update the mesh.
- The constant C_{tr} generally grows exponentially with $\Delta\underline{\ell} = \overline{\ell} - \underline{\ell}$, sometimes also involving linear terms, *i.e.* $\hat{\gamma} = 1$. Assume that for the choice of $\Delta\underline{\ell}$ at hand, we have $C_{\mathrm{tr}} \sim C_{\mathrm{ref}}$, then we have the following cases:
  - $\theta < 2\hat{\gamma} + 1$. This is for example the case of $\hat{\gamma} = 1$ using the upwind scheme $\theta = 1$. The error of the reference scheme dominates the perturbation introduced by the adaptive scheme $\|\overline{\boldsymbol{u}}_{\overline{\ell}}(T) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(T)\| \leq (C_{\mathrm{ref}}\Delta x^{\theta} + C_{\mathrm{MR}}\frac{\lambda}{\Delta x}\epsilon)T$, where if the final time horizon T. In accordance with [Cohen et al., 2003], we would like to have a thresholding error of the same order as the reference error, thus $\epsilon \sim \Delta x^{\theta+1}$. This is the same result as [Hovhannisyan and Müller, 2010, Corollary 5.2]. Observe that even if C_{tr} could be very large since depending exponentially on $\Delta\underline{\ell}$, we can always take Δx small enough to say that this term is dominated by $C_{\mathrm{ref}}T\Delta x^{\theta}$. This setting can be preferred because the good properties (namely the high order θ) are preserved.
  - $\theta = 2\hat{\gamma} + 1$. This is for example the case of $\hat{\gamma} = 0$ using the upwind scheme $\theta = 1$. The error of the reference scheme and the perturbation order are comparable. This is the regime where interesting coupling phenomena between the poor behavior of the reference numerical scheme (for example, numerical dissipation) and the mesh adaptation *via* multiresolution are possible. We have $\|\overline{\boldsymbol{u}}_{\overline{\ell}}(T) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(T)\| \leq ((C_{\mathrm{ref}} + C_{\mathrm{tr}})\Delta x^{\theta} + C_{\mathrm{MR}}\frac{\lambda}{\Delta x}\epsilon)T$. The same conclusions hold for ϵ but one might need a more precise discussion of the relative weight between C_{ref} and C_{tr}.
  - $\theta > 2\hat{\gamma} + 1$. This is for example the case of $\hat{\gamma} = 0$ using the Lax-Wendroff scheme or the OS-3 scheme: $\theta = 2$ or $\theta = 3$. The perturbation introduced by the adaptive scheme dominates the error of the reference scheme. Therefore, multiresolution introduces a huge perturbation that yields a different convergence rate. We have $\|\overline{\boldsymbol{u}}_{\overline{\ell}}(T) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(T)\| \leq (C_{\mathrm{tr}}\Delta x^{2\hat{\gamma}+1} + C_{\mathrm{MR}}\frac{\lambda}{\Delta x}\epsilon)T$, thus one has to take $\epsilon \sim \Delta x^{2\hat{\gamma}+2}$. This is the setting presented in [Duarte, 2011], where a control of the total error essentially by ϵ is acceptable in spite of the perturbation concerning the truncation error.
- Taking $\epsilon = 0$, we find the error bound of the reference scheme because the adaptive mesh will always be the uniform one at the finest level, thus the scheme degenerate into the reference scheme.

Proof of Theorem 4.5.1. The proof proceeds analogously to the ones of [Cohen et al., 2003, Proposition 4.2] and [Hovhannisyan and Müller, 2010, Theorem 5.1]. The major difference is that we do not need to assume the Harten heuristics [Cohen et al., 2003] or the reliability condition [Hovhannisyan and Müller, 2010]. Before proceeding, let us recall that $\mathbf{E}$ is the reference scheme (4.3), $\mathbf{E}_{\tilde{\Lambda}(t)}$ is the adaptive scheme on the leaves (4.9) and $\overline{\mathbf{E}}_{\tilde{\Lambda}(t)}$ is the Harten-like scheme (4.10).

$$
\begin{aligned}
\overbrace{\|\overline{\mathbf{v}}(t) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t)\|}^{\text{called } a_n \text{ in [Cohen et al., 2003]}} &= \|\mathbf{E}\overline{\mathbf{v}}_{\overline{\ell}}(t-\Delta t) - A_{\Lambda(t)}\mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| \leq \|\mathbf{E}\overline{\mathbf{v}}_{\overline{\ell}}(t-\Delta t) - \mathbf{E}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| \\
&\quad + \underbrace{\|\mathbf{E}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t) - \mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\|}_{\text{called } c_n \text{ in [Cohen et al., 2003]}} + \underbrace{\|\mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t) - A_{\Lambda(t)}\mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\|}_{\text{called } t_n \text{ in [Cohen et al., 2003]}} \\
&\leq \|\mathbf{E}\overline{\mathbf{v}}_{\overline{\ell}}(t-\Delta t) - \mathbf{E}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| + \|\mathbf{E}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t) - \overline{\mathbf{E}}_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| + \|\overline{\mathbf{E}}_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t) - \overline{\mathbf{E}}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| \\
&\quad + \|\overline{\mathbf{E}}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t) - \mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| + \|\mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t) - A_{\Lambda(t)}\mathbf{E}_{\tilde{\Lambda}(t)}A_{\tilde{\Lambda}(t)}\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| \\
&\leq \|\overline{\mathbf{v}}(t-\Delta t) - \hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\| + C_{\mathrm{tr}}\Delta t\Delta x^{2\hat{\gamma}+1} + C^1_{\mathrm{MR}}\epsilon + C^2_{\mathrm{MR}}\epsilon + C^1_{\mathrm{MR}}\epsilon.
\end{aligned}
$$

In this order, the terms are controlled by stability of the reference scheme; by truncation error, providing $\Delta x^{2\hat{\gamma}+1}$, with constant $C_{\mathrm{tr}}(\tilde{\Lambda}(t), (\phi_\alpha)_\alpha, \lambda, \hat{\gamma}, V)$; by stability condition of the Harten-like scheme and error control for the multiresolution with constant $C^1_{\mathrm{MR}}(\gamma)$; by the fact that $\mathbf{E}_{\tilde{\Lambda}(t)}$ is obtained by $\overline{\mathbf{E}}_{\tilde{\Lambda}(t)}$ averaging on the leaves followed

by a reconstruction, thus yielding a constant $C^2_{\mathrm{MR}}(\tilde{\Lambda}(t),(\phi_\alpha)_\alpha,\lambda,\hat{\gamma},\gamma,V)$; by error control for the multiresolution with constant $C^1_{\mathrm{MR}}(\gamma)$.

We would like to find constants independent of the particular tree $\tilde{\Lambda}(t)$ and thus of the time and the solution used to update the mesh. This is done by taking

$$C_{\mathrm{tr}}(\overline{\ell}-\underline{\ell},(\phi_\alpha)_\alpha,\lambda,\hat{\gamma},V)=\sup\{C_{\mathrm{tr}}(\Lambda,(\phi_\alpha)_\alpha,\lambda,\hat{\gamma},V)\ :\ \Lambda \text{ is a tree with levels } \ell\in[\![\underline{\ell},\overline{\ell}]\!]\}.$$

Since the truncation errors generally grow with $\Delta\ell$, normally $C_{\mathrm{tr}}=C_{\mathrm{tr}}(\Lambda_{\underline{\ell}},(\phi_\alpha)_\alpha,\lambda,\hat{\gamma},V)$, where $\Lambda_{\underline{\ell}}$ is the tree corresponding to a uniform mesh at level $\underline{\ell}$, that is $\Lambda_{\underline{\ell}}=\{(\underline{\ell},k)\ :\ k\in[\![0,N_{\underline{\ell}}[\![\}$. We use the same procedure to deal with C^2_{MR}, thus gathering all the constants gives Then $\|\overline{\mathbf{v}}(t)-\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t)\|\leq\|\overline{\mathbf{v}}_{\overline{\ell}}(t-\Delta t)-\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(t-\Delta t)\|+C_{\mathrm{tr}}\Delta t\Delta x^{2\hat{\gamma}+1}+C_{\mathrm{MR}}\epsilon$, thus iterating concludes the proof. □

4.6 Numerical tests

Before concluding, we would like to check that information provided by the modified equation that we have previously derived describes the actual behavior of the adaptive numerical schemes. Contrarily to Section 3.1, we really employ adaptive meshes which move in time.

4.6.1 Convergence study

We consider the domain $\Omega=[-2,2]$ and a final time for the simulation $T=1/2$. The transport velocity is set to $V=1$ and the Courant number is $\lambda=2$. The initial datum is $u^\circ(x)=\exp(1/(x^2-1))\mathbb{1}_{[-1,1]}(x)$, which is infinitely continuously differentiable with compact support. Multiresolution for adapting the computational mesh takes $\epsilon=1\mathrm{e}{-3}$ and $\gamma=1$. This choice of threshold has to be compared with the initial datum since at the beginning of the simulation $|\overline{\mathrm{d}}_{\ell,k}(t=0)|\lesssim 2^{-3\ell}|u^\circ|_{W^{3,\infty}(\mathbb{R})}\simeq 186.4\times 2^{-3\ell}$, *cf.* Proposition 2.3.2. We therefore deduce that the actual level of the mesh in the zones where the previous $W^{3,\infty}$-semi-norm is attained is $\ell\simeq(\overline{\ell}-\log_2(10^{-3}/186.4))/4$. This means that starting from $\overline{\ell}$ between 5 and 6, there will be no zone in the initial mesh to be refined at the finest level. The reader can verify that this is actually the case when numerical results are presented.

We employ the upwind, the Lax-Wendroff and OS-3 as reference schemes, adapted using $\hat{\gamma}=0$ and $\hat{\gamma}=1$. We indicate $\overline{\ell}^{\mathrm{end}}$ the maximum level of resolution present in the mesh at the end of the numerical simulation, since as previously discussed when introducing the choice of ϵ, the adaptive mesh does not necessarily reach the finest level and we employ a fully adaptive method, so the solution does not need to be encoded up to level $\overline{\ell}$. We consider the following L^1 quantities, computed at the final time of the simulation:

- $E_{\mathrm{ref}}=\|\overline{\boldsymbol{u}}_{\overline{\ell}}(T)-\overline{\mathbf{v}}(T)\|_{\ell^1}$, the error of the reference scheme. We expect $E_{\mathrm{ref}}\sim\Delta x^\theta$.
- $E^{\overline{\ell}}_{\mathrm{adap}}=\|\overline{\boldsymbol{u}}_{\overline{\ell}}(T)-\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(T)\|_{\ell^1}$, the error of the adaptive scheme. It depends on the scheme, $\underline{\ell}$, $\overline{\ell}$, the way of computing the fluxes $\hat{\gamma}$, γ and ϵ.
- $D_{\mathrm{adap}}=\|\overline{\mathbf{v}}(T)-\hat{\hat{\mathbf{w}}}_{\overline{\ell}}(T)\|_{\ell^1}$, difference between reference and adaptive solution. It has the same dependencies as $E^{\overline{\ell}}_{\mathrm{adap}}$. We expect the bound $D_{\mathrm{adap}}\lesssim C_{\mathrm{tr}}\Delta x^{2\hat{\gamma}+1}+C_{\mathrm{MR}}\frac{\lambda\epsilon}{\Delta x}$.
- $\hat{D}_{\mathrm{adap}}$, estimator of D_{adap} based on the local truncation error from the modified equations. The modified equation for the adaptive scheme at distance $\Delta\ell$ from the finest level being

$$\partial_t u+V\partial_x u=\sum_{h=2}^{+\infty}\Delta x^{h-1}\sigma_h^{\Delta\ell}\partial_x^h u,$$

  we have shown in Theorem 4.3.1 and Corollary 4.4.1 that $\sigma_h^{\Delta\ell}$ is independent of $\Delta\ell$ for $h\in[\![2,2\hat{\gamma}+1]\!]$. Let us emphasize that the derivative $\partial_x^{2\gamma+2}$ cannot be accurately estimated using the multiresolution transform used for the mesh adaptation because this derivative is not controlled by the procedure (it is indeed the first one not being controlled) and would yield extremely noisy results. For this reasons, we shall utilize the exact

solution of the problem instead. We set

$$\hat{D}_{\text{adap}} := \sum_{(\ell,k)\in S(\tilde{\Lambda}(T))} |\sigma^{\Delta\ell}_{2\hat{\gamma}+2} - \sigma_{2\hat{\gamma}+2}| \sum_{r=0}^{2^{\Delta\ell}-1} \Delta x^{2\hat{\gamma}+2} |\partial_x^{2\hat{\gamma}+2} u(T, x_{\overline{\ell},2^{\Delta\ell}k+r})|,$$

which is a sort of L^1 estimation of the truncation error. Observe that the following bias are present in such estimator:

- It neglects the influence of the mesh adaptation at each step *via* ϵ. However, this contribution is frequently over-estimated, because the estimations bound all the non-significant (small) details—which make up the mesh adaptation error—by the threshold ϵ but these details are very often way smaller than the threshold [Cohen et al., 2003].
- It considers that the adaptive mesh does not change in time, but only translates at velocity V. This is essentially true if the scheme is of relatively high order, so that the numerical solution is very close to the exact solution.
- It uses the $2\hat{\gamma}+2$ derivative of the exact solution, instead of that of the numerical solution of the scheme. We could use an estimation with Finite Differences on the solution reconstructed at the finest level. However, in the case where the $\hat{\gamma} = \gamma$ the computations are—as previously observed—extremely noisy and inaccurate, thus unreliable. We therefore decided to estimate using the analytic solution, because ultimately the role of this term is just to provide a sort of "weighting measure" in the integral.

We shall be interested in the numerical order of convergence of $\hat{D}_{\text{adap}}$, because we neglect fixed constants. According to the modified equations that we have previously provided for the three schemes at hand, we consider

- Upwind and Lax-Wendroff scheme

$$|\sigma^{\Delta\ell}_{2\hat{\gamma}+2} - \sigma_{2\hat{\gamma}+2}| = \begin{cases} |2^{\Delta\ell} - 1|, & \text{for} \quad \hat{\gamma} = 0, \\ |-3\Delta\ell 2^{2\Delta\ell} + 2^{2\Delta\ell} - 1|, & \text{for} \quad \hat{\gamma} = 1. \end{cases}$$

- OS-3 scheme

$$|\sigma^{\Delta\ell}_{2\hat{\gamma}+2} - \sigma_{2\hat{\gamma}+2}| = \begin{cases} \left| 2^{\Delta\ell}\left(-\frac{V^2}{\lambda^2} + 2\frac{V}{\lambda} + 1\right) - 3\frac{V}{\lambda} \right|, & \text{for} \quad \hat{\gamma} = 0, \\ \left| (2^{2\Delta\ell} - 1)\left(2\frac{V^2}{\lambda^2} + \frac{V}{\lambda^2} - 2\right) - 3\Delta\ell 2^{2\Delta\ell}\frac{V}{\lambda} \right|, & \text{for} \quad \hat{\gamma} = 1. \end{cases}$$

- $E^{\overline{\ell}}_{\text{coa}}$, error of an adaptive scheme using a uniform mesh of level $\overline{\ell}^{\text{end}}$.
- D_{coa}, difference between the solution of the reference scheme and the solution of an adaptive scheme using a uniform mesh at level $\overline{\ell}^{\text{end}}$.

Table 4.1: Upwind. Errors for $\hat{\gamma} = 0$. Minimum level $\underline{\ell}$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	E_{ref} [order]	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
2	4	4	0	2.18E-02 [—-]	2.23E-02 [—-]	4.55E-04 [—-]		2.18E-02 [—-]	0.00E+00 [—-]
2	5	5	0	1.16E-02 [0.91]	2.02E-02 [0.14]	8.67E-03 [-4.25]		1.16E-02 [0.91]	0.00E+00 [—-]
2	6	5	1	6.01E-03 [0.95]	2.22E-02 [-0.14]	1.62E-02 [-0.91]	-1.07	1.69E-02 [-0.54]	1.09E-02 [—-]
2	7	6	1	3.06E-03 [0.97]	1.93E-02 [0.20]	1.63E-02 [0.00]	0.04	8.85E-03 [0.93]	5.81E-03 [0.91]
2	8	6	2	1.54E-03 [0.99]	1.63E-02 [0.24]	1.48E-02 [0.14]	0.27	1.02E-02 [-0.21]	8.71E-03 [-0.58]
2	9	6	3	7.76E-04 [0.99]	1.54E-02 [0.08]	1.47E-02 [0.01]	0.02	1.09E-02 [-0.09]	1.02E-02 [-0.22]
2	10	6	4	3.89E-04 [1.00]	1.36E-02 [0.18]	1.32E-02 [0.15]	0.06	1.13E-02 [-0.04]	1.09E-02 [-0.10]
2	11	7	4	1.95E-04 [1.00]	1.05E-02 [0.38]	1.03E-02 [0.36]	0.53	5.83E-03 [0.95]	5.63E-03 [0.95]
2	12	7	5	9.74E-05 [1.00]	8.89E-03 [0.23]	8.79E-03 [0.22]	0.20	5.92E-03 [-0.02]	5.82E-03 [-0.05]
2	13	7	6	4.87E-05 [1.00]	7.83E-03 [0.18]	7.78E-03 [0.18]	0.19	5.96E-03 [-0.01]	5.91E-03 [-0.02]

For the upwind scheme, fixing $\underline{\ell} = 2$, we obtain the result in Table 4.1 for $\hat{\gamma} = 0$. The reference scheme gives the expected order $\theta = 1$. Very early, the fact that the level jump with respect to $\overline{\ell}$ in the mesh increases, due to

the regularity of the solution, the fact that ϵ is fixed and $2\hat{\gamma}+1=1$, gives that $D_{\text{adap}} \gg E_{\text{ref}}$, thus $E^{\overline{\ell}}_{\text{adap}} \sim D_{\text{adap}}$. The numerical order for $E^{\overline{\ell}}_{\text{adap}}$ and D_{adap} is not clear, since the actual error is expected to behave way differently than the upper bound from Theorem 4.5.1, assuming that the $\overline{\ell}-\underline{\ell}$ is fixed. Still, we see that we can use $\hat{D}_{\text{adap}}$ to understand the trend of D_{adap}, observing small discrepancies between the numerical orders. This validates the use of the modified equations to understand the behavior of the truncation error of the adaptive method. Concerning $E^{\overline{\ell}}_{\text{coa}}$ and D_{coa}, we see that we obtain the expected linear rate in Δx uniquely when $\Delta\overline{\ell}^{\text{end}}$ does not change when increasing $\overline{\ell}$ by one. We also see that $E^{\overline{\ell}}_{\text{coa}} \sim E^{\overline{\ell}}_{\text{adap}}$ and $D_{\text{coa}} \sim D_{\text{adap}}$, which means that the local error coming from the cells at the finest level of resolution attained in the adaptive mesh explains most of the overall error. Moreover, the effect of the thresholding ϵ is moderate in the considered setting.

Table 4.2: Upwind. Errors for $\hat{\gamma}=1$. Minimum level $\underline{\ell}$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
2	4	4	0	2.18E-02 [—]	3.51E-05 [—]		2.18E-02 [—]	0.00E+00 [—]
2	5	5	0	1.16E-02 [0.91]	8.98E-05 [-1.35]	-0.91	1.16E-02 [0.91]	0.00E+00 [—]
2	6	5	1	6.12E-03 [0.93]	2.50E-04 [-1.48]	-4.08	6.12E-03 [0.92]	2.11E-04 [—]
2	7	6	1	3.11E-03 [0.98]	1.21E-04 [1.04]	2.44	3.07E-03 [0.99]	3.96E-05 [2.41]
2	8	6	2	1.58E-03 [0.98]	8.34E-05 [0.54]	0.26	1.57E-03 [0.97]	5.88E-05 [-0.57]
2	9	6	3	8.03E-04 [0.97]	6.17E-05 [0.44]	0.44	8.03E-04 [0.96]	5.89E-05 [0.00]
2	10	7	3	4.06E-04 [0.98]	3.29E-05 [0.91]	1.88	3.91E-04 [1.04]	8.61E-06 [2.78]
2	11	7	4	2.01E-04 [1.01]	1.63E-05 [1.01]	1.22	1.97E-04 [0.99]	6.85E-06 [0.33]
2	12	7	5	1.01E-04 [1.00]	8.66E-06 [0.92]	1.01	9.94E-05 [0.99]	5.28E-06 [0.38]
2	13	8	5	5.11E-05 [0.98]	5.33E-06 [0.70]	1.03	4.89E-05 [1.03]	6.10E-07 [3.11]

For $\hat{\gamma}=1$, the results are given in Table 4.2. Compared to the case $\hat{\gamma}=0$, we observe that the adaptive mesh is often refined more finely, which is probably due to the reduced numerical dissipation (only the one due to the reference upwind scheme). Since now $D_{\text{adap}} \ll E_{\text{ref}}$, thanks to the fact that we have $2\hat{\gamma}+1=3>1$, we have that $E^{\overline{\ell}}_{\text{adap}} \sim E_{\text{ref}}$, which also shows that the role played by ϵ is marginal. The trend for D_{adap} seems to be essentially linear in Δx, especially for large $\overline{\ell}$. It can be explained by the following way of reasoning. For $\Delta\ell \gg 1$, which is the case when we increase $\overline{\ell}$ in the test case, we have $|\sigma_4^{\Delta\ell}-\sigma_4^0| \sim \Delta\ell 2^{2\Delta\ell}$. Assume that when we increase $\overline{\ell}$ by one, we also increase $\Delta\overline{\ell}^{\text{end}}$ by one, then we can estimate the convergence rate by $\frac{\Delta\overline{\ell}^{\text{end}} 2^{2\Delta\overline{\ell}^{\text{end}}}\Delta x}{(\Delta\overline{\ell}^{\text{end}}+1)2^{2(\Delta\overline{\ell}^{\text{end}}+1)}\Delta x^3/8} = 2\frac{\Delta\overline{\ell}^{\text{end}}}{\Delta\overline{\ell}^{\text{end}}+1} \sim 2$, thus resulting in the linear rate. This again shows that the analysis with the modified equations provides a good insight into the actual behavior of the scheme. The trend for $\hat{D}_{\text{adap}}$ is similar to the one of D_{adap} except for some tests, where some discrepancies appear. Finally, looking at D_{coa}, we see that in the cases where $\Delta\overline{\ell}^{\text{end}}$ does not change when $\overline{\ell}$ increases, we observe third-order convergence in Δx, since $2\hat{\gamma}+1=3$.

Table 4.3: Upwind. Errors for $\hat{\gamma}=0$. Having $\overline{\ell}-\underline{\ell}=3$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
1	4	4	0	2.23E-02 [—]	4.55E-04 [—]		2.18E-02 [—]	0.00E+00 [—]
2	5	5	0	2.02E-02 [0.14]	8.67E-03 [-4.25]		1.16E-02 [0.91]	0.00E+00 [—]
3	6	5	1	2.22E-02 [-0.14]	1.62E-02 [-0.91]	-1.07	1.69E-02 [-0.54]	1.09E-02 [—]
4	7	6	1	2.13E-02 [0.06]	1.82E-02 [-0.17]	-0.03	8.85E-03 [0.93]	5.81E-03 [0.91]
5	8	6	2	1.57E-02 [0.44]	1.41E-02 [0.37]	0.44	1.02E-02 [-0.21]	8.71E-03 [-0.58]
6	9	6	3	1.07E-02 [0.55]	9.95E-03 [0.51]	0.45	1.09E-02 [-0.09]	1.02E-02 [-0.22]
7	10	7	3	5.64E-03 [0.92]	5.26E-03 [0.92]	1.00	5.64E-03 [0.95]	5.26E-03 [0.95]
8	11	8	3	2.87E-03 [0.97]	2.68E-03 [0.97]	1.00	2.87E-03 [0.97]	2.68E-03 [0.97]
9	12	9	3	1.45E-03 [0.99]	1.35E-03 [0.99]	1.00	1.45E-03 [0.99]	1.35E-03 [0.99]
10	13	10	3	7.28E-04 [0.99]	6.79E-04 [0.99]	1.00	7.28E-04 [0.99]	6.79E-04 [0.99]

We now repeat the same test for $\hat{\gamma}=0$ but we fix $\overline{\ell}-\underline{\ell}=3$. The result are given in Table 4.3. We see that $E^{\overline{\ell}}_{\text{adap}} \gg E_{\text{ref}}$ since $D_{\text{adap}} \gg E_{\text{ref}}$ but except at the very beginning, we observe the right convergence rate. This is due to the fact that for large $\Delta\ell$ the adaptive computational mesh is indeed a uniform mesh at level $\overline{\ell}-3$, fact which is also confirmed by looking at $E^{\overline{\ell}}_{\text{coa}}$ and D_{coa}. In this case, we perfectly fall into the framework where the bounds from Theorem 4.5.1 (and Section 3.1) give the trend of the actual errors, also with no error coming from the mesh adaptation because the mesh does not evolve since it remains uniform at some coarse level of resolution. We observe that also for small $\overline{\ell}$, the order of $\hat{D}_{\text{adap}}$ yields very good results.

Table 4.4: Upwind. Errors for $\hat{\gamma} = 1$. Having $\overline{\ell} - \underline{\ell} = 3$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
1	4	4	0	2.18E-02 [—-]	3.46E-05 [—-]		2.18E-02 [—-]	0.00E+00 [—-]
2	5	5	0	1.16E-02 [0.91]	8.98E-05 [-1.37]	-0.91	1.16E-02 [0.91]	0.00E+00 [—-]
3	6	5	1	6.12E-03 [0.93]	2.50E-04 [-1.48]	-4.08	6.12E-03 [0.92]	2.11E-04 [—-]
4	7	6	1	3.10E-03 [0.98]	1.21E-04 [1.04]	2.44	3.07E-03 [0.99]	3.96E-05 [2.41]
5	8	6	2	1.58E-03 [0.98]	7.60E-05 [0.67]	0.31	1.57E-03 [0.97]	5.88E-05 [-0.57]
6	9	6	3	7.99E-04 [0.98]	5.23E-05 [0.54]	0.51	8.03E-04 [0.96]	5.89E-05 [0.00]
7	10	7	3	3.91E-04 [1.03]	8.61E-06 [2.60]	3.00	3.91E-04 [1.04]	8.61E-06 [2.78]
8	11	8	3	1.95E-04 [1.01]	1.19E-06 [2.85]	3.00	1.95E-04 [1.01]	1.19E-06 [2.85]
9	12	9	3	9.74E-05 [1.00]	1.58E-07 [2.92]	3.00	9.74E-05 [1.00]	1.58E-07 [2.92]
10	13	10	3	4.87E-05 [1.00]	2.03E-08 [2.96]	3.00	4.87E-05 [1.00]	2.03E-08 [2.96]

For the case $\hat{\gamma} = 1$ with fixed $\overline{\ell} - \underline{\ell} = 3$, see Table 4.4, the results are comparable with those of Table 4.3, with the fact that now $2\hat{\gamma} + 1 = 3$, thus orders are different.

Table 4.5: Lax-Wendroff. Errors for $\hat{\gamma} = 0$. Minimum level $\underline{\ell}$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	E_{ref} [order]	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
2	4	4	0	6.52E-03 [—-]	7.16E-03 [—-]	8.01E-04 [—-]		6.52E-03 [—-]	0.00E+00 [—-]
2	5	5	0	1.95E-03 [1.74]	6.68E-03 [0.10]	4.88E-03 [-2.61]		1.95E-03 [1.74]	0.00E+00 [—-]
2	6	6	0	5.49E-04 [1.83]	7.70E-03 [-0.20]	7.35E-03 [-0.59]	-0.74	5.49E-04 [1.83]	0.00E+00 [—-]
2	7	6	1	1.41E-04 [1.96]	8.31E-03 [-0.11]	8.30E-03 [-0.18]	-0.25	3.10E-03 [-2.50]	3.10E-03 [—-]
2	8	6	2	3.45E-05 [2.03]	7.78E-03 [0.09]	7.78E-03 [0.09]	0.22	4.57E-03 [-0.56]	4.57E-03 [-0.56]
2	9	7	2	8.53E-06 [2.02]	7.61E-03 [0.03]	7.61E-03 [0.03]	0.01	2.31E-03 [0.98]	2.31E-03 [0.98]
2	10	7	3	2.13E-06 [2.00]	6.61E-03 [0.20]	6.61E-03 [0.20]	0.11	2.69E-03 [-0.22]	2.69E-03 [-0.22]
2	11	7	4	5.31E-07 [2.00]	5.22E-03 [0.34]	5.22E-03 [0.34]	0.54	2.87E-03 [-0.10]	2.87E-03 [-0.10]
2	12	7	5	1.33E-07 [2.00]	4.52E-03 [0.21]	4.52E-03 [0.21]	0.16	2.97E-03 [-0.05]	2.97E-03 [-0.05]
2	13	8	5	3.32E-08 [2.00]	4.05E-03 [0.16]	4.05E-03 [0.16]	0.16	1.50E-03 [0.99]	1.50E-03 [0.99]

Table 4.6: Lax-Wendroff. Errors for $\hat{\gamma} = 1$. Minimum level $\underline{\ell}$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
2	4	4	0	6.53E-03 [—-]	3.74E-05 [—-]		6.52E-03 [—-]	0.00E+00 [—-]
2	5	5	0	1.98E-03 [1.72]	7.18E-05 [-0.94]	-1.28	1.95E-03 [1.74]	0.00E+00 [—-]
2	6	6	0	5.97E-04 [1.73]	1.71E-04 [-1.25]	-1.87	5.49E-04 [1.83]	0.00E+00 [—-]
2	7	6	1	2.00E-04 [1.58]	1.09E-04 [0.65]	0.50	1.59E-04 [1.78]	4.96E-05 [—-]
2	8	6	2	9.99E-05 [1.00]	8.31E-05 [0.39]	0.31	7.75E-05 [1.04]	5.89E-05 [-0.25]
2	9	7	2	4.95E-05 [1.01]	4.63E-05 [0.84]	0.74	1.26E-05 [2.62]	7.27E-06 [3.02]
2	10	7	3	2.76E-05 [0.84]	2.76E-05 [0.74]	1.64	7.12E-06 [0.82]	6.27E-06 [0.21]
2	11	7	4	1.39E-05 [0.99]	1.38E-05 [1.01]	1.25	5.27E-06 [0.44]	5.07E-06 [0.31]
2	12	7	5	7.98E-06 [0.80]	7.92E-06 [0.80]	0.98	4.39E-06 [0.26]	4.33E-06 [0.23]
2	13	8	5	4.72E-06 [0.76]	4.72E-06 [0.75]	0.98	4.07E-07 [3.43]	3.96E-07 [3.45]

We repeat the same tests for the Lax-Wendroff scheme using $\hat{\gamma} = 0$, see Table 4.5, and $\hat{\gamma} = 1$, see Table 4.6, testing for fixed $\underline{\ell}$. The results allow to draw the same conclusions as the test for the upwind scheme. However, due to $\theta = 2$, we observe that rapidly $D_{\text{adap}} \gg E_{\text{ref}}$, hence driving the evolution of $E^{\overline{\ell}}_{\text{adap}}$. This is due to the fact that ϵ is kept fixed when reducing the step size, which both adds error from the truncation error (see $\Delta\overline{\ell}^{\text{end}}$ increasing) and from the thresholding of the mesh.

We therefore perform the same test with $\hat{\gamma} = 1$ but varying ϵ as $\overline{\ell}$ increases. In particular, we test $\epsilon \sim \Delta x$ and $\epsilon \sim \Delta x^3$, as previously suggested. The results are given in Table 4.7 and indeed show better convergence rates for D_{adap} and thus less severe perturbations. In particular, for the choice $\epsilon \sim \Delta x^3$, we see that $D_{\text{adap}} \ll E_{\text{ref}}$, thus $E^{\overline{\ell}}_{\text{adap}} \sim E_{\text{ref}}$. Once again, $\hat{D}_{\text{adap}}$ is a robust estimator of D_{adap}.

Repeating the same test for OS-3 with fixed ϵ, see Table 4.8 and Table 4.9, we obtain results comparable to the ones for the Lax-Wendroff scheme. Again, this calls for the reduction of ϵ with Δx in order to preserve the accuracy of the reference scheme, now with $\theta = 3$.

Hence, taking $\hat{\gamma} = 1$, we vary $\epsilon \sim \Delta x$ and $\epsilon \sim \Delta x^4$, see Table 4.10. The first choice is not enough to preserve the order of the method, as one could expect, since rapidly $D_{\text{adap}} \gg E_{\text{ref}}$. On the other hand, we see that the second choice guarantees to preserve the order of convergence. It must be noted that reducing ϵ according to the space

Table 4.7: Lax-Wendroff. Errors for $\hat{\gamma} = 1$. Minimum level $\underline{\ell}$ fixed. The threshold ϵ decreases with Δx (first half of the table) and with Δx^3 (second half of the table).

ϵ	$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
4.00E-04	2	4	4	0	6.64E-03 [—-]	3.29E-04 [—-]		6.52E-03 [—-]	0.00E+00 [—-]
2.00E-04	2	5	5	0	2.00E-03 [1.73]	1.61E-04 [1.03]	0.06	1.95E-03 [1.74]	0.00E+00 [—-]
1.00E-04	2	6	6	0	5.97E-04 [1.75]	1.71E-04 [-0.09]	-0.56	5.49E-04 [1.83]	0.00E+00 [—-]
5.00E-05	2	7	6	1	1.76E-04 [1.76]	7.48E-05 [1.19]	0.85	1.59E-04 [1.78]	4.96E-05 [—-]
2.50E-05	2	8	7	1	4.36E-05 [2.01]	2.82E-05 [1.41]	1.54	3.61E-05 [2.14]	6.37E-06 [2.96]
1.25E-05	2	9	7	2	1.53E-05 [1.51]	1.05E-05 [1.42]	1.42	1.26E-05 [1.52]	7.27E-06 [-0.19]
6.25E-06	2	10	8	2	4.15E-06 [1.89]	3.50E-06 [1.59]	1.56	2.41E-06 [2.38]	8.81E-07 [3.04]
3.13E-06	2	11	8	3	1.46E-06 [1.51]	1.23E-06 [1.50]	1.76	9.71E-07 [1.31]	7.16E-07 [0.30]
1.56E-06	2	12	9	3	4.92E-07 [1.57]	4.65E-07 [1.41]	1.64	1.66E-07 [2.54]	8.57E-08 [3.06]
7.81E-07	2	13	9	4	1.45E-07 [1.77]	1.34E-07 [1.80]	1.84	7.29E-08 [1.19]	6.05E-08 [0.50]
6.40E-03	2	4	3	1	1.25E-02 [—-]	8.11E-03 [—-]		1.17E-02 [—-]	7.42E-03 [—-]
8.00E-04	2	5	4	1	3.23E-03 [1.95]	2.07E-03 [1.97]	2.84	3.00E-03 [1.97]	1.81E-03 [2.04]
1.00E-04	2	6	6	0	5.97E-04 [2.44]	1.71E-04 [3.60]	5.03	5.49E-04 [2.45]	0.00E+00 [—-]
1.25E-05	2	7	7	0	1.42E-04 [2.07]	2.22E-05 [2.95]	2.69	1.41E-04 [1.96]	0.00E+00 [—-]
1.56E-06	2	8	8	0	3.41E-05 [2.06]	3.34E-06 [2.73]	2.71	3.45E-05 [2.03]	0.00E+00 [—-]
1.95E-07	2	9	9	0	8.41E-06 [2.02]	5.31E-07 [2.65]	2.68	8.53E-06 [2.02]	0.00E+00 [—-]
2.44E-08	2	10	10	0	2.10E-06 [2.00]	9.01E-08 [2.56]	2.91	2.13E-06 [2.00]	0.00E+00 [—-]
3.05E-09	2	11	11	0	5.25E-07 [2.00]	1.46E-08 [2.63]	2.83	5.31E-07 [2.00]	0.00E+00 [—-]
3.81E-10	2	12	12	0	1.32E-07 [2.00]	2.25E-09 [2.69]	2.74	1.33E-07 [2.00]	0.00E+00 [—-]
4.77E-11	2	13	13	0	3.29E-08 [2.00]	4.63E-10 [2.28]	2.92	3.32E-08 [2.00]	0.00E+00 [—-]

Table 4.8: OS-3. Errors for $\hat{\gamma} = 0$. Minimum level $\underline{\ell}$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	E_{ref} [order]	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
2	4	4	0	2.31E-03 [—-]	2.58E-03 [—-]	3.16E-04 [—-]		2.31E-03 [—-]	0.00E+00 [—-]
2	5	5	0	4.77E-04 [2.28]	8.71E-03 [-1.76]	8.29E-03 [-4.71]		4.77E-04 [2.28]	0.00E+00 [—-]
2	6	6	0	8.07E-05 [2.56]	1.21E-02 [-0.48]	1.21E-02 [-0.54]	-0.60	8.07E-05 [2.56]	0.00E+00 [—-]
2	7	6	1	1.14E-05 [2.82]	1.38E-02 [-0.18]	1.38E-02 [-0.19]	-0.45	6.01E-03 [-6.22]	6.01E-03 [—-]
2	8	6	2	1.50E-06 [2.93]	1.21E-02 [0.18]	1.21E-02 [0.18]	0.33	7.45E-03 [-0.31]	7.45E-03 [-0.31]
2	9	7	2	1.88E-07 [2.99]	1.14E-02 [0.09]	1.14E-02 [0.09]	0.03	3.81E-03 [0.97]	3.81E-03 [0.97]
2	10	7	3	2.35E-08 [3.00]	9.92E-03 [0.20]	9.92E-03 [0.20]	0.17	4.18E-03 [-0.13]	4.18E-03 [-0.13]
2	11	7	4	2.94E-09 [3.00]	7.85E-03 [0.34]	7.85E-03 [0.34]	0.53	4.36E-03 [-0.06]	4.36E-03 [-0.06]
2	12	7	5	3.67E-10 [3.00]	6.76E-03 [0.22]	6.76E-03 [0.22]	0.12	4.46E-03 [-0.03]	4.46E-03 [-0.03]
2	13	8	5	4.59E-11 [3.00]	5.95E-03 [0.18]	5.95E-03 [0.18]	0.19	2.26E-03 [0.98]	2.26E-03 [0.98]

Table 4.9: OS-3. Errors for $\hat{\gamma} = 1$. Minimum level $\underline{\ell}$ fixed.

$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
2	4	4	0	2.35E-03 [—-]	5.19E-05 [—-]		2.31E-03 [—-]	0.00E+00 [—-]
2	5	5	0	5.67E-04 [2.05]	9.92E-05 [-0.94]	-1.28	4.77E-04 [2.28]	0.00E+00 [—-]
2	6	6	0	2.95E-04 [0.94]	2.20E-04 [-1.15]	-1.30	8.07E-05 [2.56]	0.00E+00 [—-]
2	7	6	1	1.66E-04 [0.83]	1.56E-04 [0.50]	0.13	9.24E-05 [-0.19]	8.12E-05 [—-]
2	8	6	2	1.02E-04 [0.70]	1.01E-04 [0.62]	0.55	7.87E-05 [0.23]	7.73E-05 [0.07]
2	9	7	2	4.90E-05 [1.06]	4.89E-05 [1.05]	1.15	1.06E-05 [2.89]	1.05E-05 [2.89]
2	10	7	3	3.01E-05 [0.70]	3.01E-05 [0.70]	1.44	7.86E-06 [0.44]	7.84E-06 [0.42]
2	11	7	4	1.47E-05 [1.03]	1.47E-05 [1.03]	1.28	5.80E-06 [0.44]	5.79E-06 [0.44]
2	12	7	5	8.07E-06 [0.87]	8.07E-06 [0.87]	1.03	4.56E-06 [0.35]	4.56E-06 [0.35]
2	13	8	5	4.88E-06 [0.73]	4.88E-06 [0.73]	1.05	4.44E-07 [3.36]	4.44E-07 [3.36]

steps has two effects. On one hand, it makes the thresholding error (proportional to ϵ) of the same order than the error of the reference scheme. On the other hand, it makes the truncation error of the adaptive scheme smaller because of the presence of more refined meshes.

4.6.2 Coupling in time

We finally want to check if the modified equations allow to understand the coupling effect between poor behavior of the multiresolution Finite Volume scheme (which hence deflates the details of the solution by a smoothing effect) and the thresholding $\mathcal{T}_\epsilon$. Let us consider the framework of the upwind scheme, hence $\theta = 1$ and use $\gamma = 1$

Table 4.10: OS-3. Errors for $\hat{\gamma} = 1$. Minimum level $\underline{\ell}$ fixed. The threshold ϵ decreases with Δx (first half of the table) and with Δx^4 (second half of the table).

ϵ	$\underline{\ell}$	$\overline{\ell}$	$\overline{\ell}^{\text{end}}$	$\Delta\overline{\ell}^{\text{end}}$	$E^{\overline{\ell}}_{\text{adap}}$ [order]	D_{adap} [order]	order $\hat{D}_{\text{adap}}$	$E^{\overline{\ell}}_{\text{coa}}$ [order]	D_{coa} [order]
4.00E-04	2	4	4	0	2.74E-03 [—-]	4.71E-04 [—-]		2.31E-03 [—-]	0.00E+00 [—-]
2.00E-04	2	5	5	0	6.32E-04 [2.11]	1.74E-04 [1.44]	0.59	4.77E-04 [2.28]	0.00E+00 [—-]
1.00E-04	2	6	6	0	2.95E-04 [1.10]	2.20E-04 [-0.34]	-0.12	8.07E-05 [2.56]	0.00E+00 [—-]
5.00E-05	2	7	6	1	1.17E-04 [1.33]	1.06E-04 [1.05]	0.44	9.24E-05 [-0.19]	8.12E-05 [—-]
2.50E-05	2	8	7	1	3.59E-05 [1.71]	3.46E-05 [1.62]	2.08	1.30E-05 [2.82]	1.16E-05 [2.81]
1.25E-05	2	9	7	2	1.43E-05 [1.33]	1.41E-05 [1.29]	1.12	1.06E-05 [0.29]	1.05E-05 [0.14]
6.25E-06	2	10	8	2	4.20E-06 [1.77]	4.18E-06 [1.76]	1.68	1.35E-06 [2.97]	1.33E-06 [2.97]
3.13E-06	2	11	8	3	1.48E-06 [1.51]	1.47E-06 [1.50]	1.81	9.52E-07 [0.51]	9.49E-07 [0.49]
1.56E-06	2	12	9	3	5.13E-07 [1.53]	5.13E-07 [1.52]	1.68	1.16E-07 [3.03]	1.16E-07 [3.03]
7.81E-07	2	13	9	4	1.49E-07 [1.79]	1.49E-07 [1.79]	1.91	7.55E-08 [0.62]	7.54E-08 [0.62]
2.56E-02	2	4	2	2	3.68E-02 [—-]	3.63E-02 [—-]		3.68E-02 [—-]	3.63E-02 [—-]
1.60E-03	2	5	4	1	3.14E-03 [3.55]	2.75E-03 [3.72]	5.63	2.65E-03 [3.80]	2.26E-03 [4.00]
1.00E-04	2	6	6	0	2.95E-04 [3.41]	2.20E-04 [3.64]	5.59	8.07E-05 [5.04]	0.00E+00 [—-]
6.25E-06	2	7	7	0	2.35E-05 [3.65]	1.30E-05 [4.08]	4.10	1.14E-05 [2.82]	0.00E+00 [—-]
3.91E-07	2	8	8	0	2.30E-06 [3.35]	8.57E-07 [3.93]	3.77	1.50E-06 [2.93]	0.00E+00 [—-]
2.44E-08	2	9	9	0	2.47E-07 [3.22]	6.26E-08 [3.77]	3.68	1.88E-07 [2.99]	0.00E+00 [—-]
1.53E-09	2	10	10	0	2.82E-08 [3.13]	5.03E-09 [3.64]	3.36	2.35E-08 [3.00]	0.00E+00 [—-]
9.54E-11	2	11	11	0	4.38E-09 [2.69]	1.47E-09 [1.77]	3.77	2.94E-09 [3.00]	0.00E+00 [—-]
5.96E-12	2	12	12	0	4.48E-10 [3.29]	8.31E-11 [4.15]	3.98	3.67E-10 [3.00]	0.00E+00 [—-]
3.73E-13	2	13	13	0	5.09E-11 [3.14]	5.19E-12 [4.00]	3.90	4.59E-11 [3.00]	0.00E+00 [—-]

with $\hat{\gamma} = 0$, thus the modified equation contains a perturbed dissipation term. We have that for a function

$$\phi_a(x) = \frac{1}{\sqrt{4\pi a}} e^{-x^2/(4a)}, \qquad \text{then} \qquad |\phi_a|_{W^{3,\infty}(\mathbb{R})} = \|\partial_x^3 \phi_a\|_{L^\infty(\mathbb{R})} = \underbrace{\frac{1}{4} e^{(\sqrt{6}-3)/2} \sqrt{\frac{3(3-\sqrt{6})}{\pi}}}_{=C_3 \simeq 0.13765} a^{-2}.$$

Considering that the solution of the numerical scheme satisfies, up to second-order, the modified equation

$$\partial_t u + V \partial_x u - \Delta x \underbrace{\frac{V}{2}\left(2^{\Delta\ell} - \frac{V}{\lambda}\right)}_{=\sigma_2^{\Delta\ell}} \partial_{xx} u = 0.$$

The initial profile to be considered is $u^\circ(x) = \frac{1}{\sqrt{4\pi\kappa}} e^{-x^2/(4\kappa)}$. Assume that at the initial time, the maximum level is attained (thanks to the choice of ϵ).

- $\Delta\ell = 0$. The solution of the modified equation would be, if the mesh were uniform

$$u(t,x) = \frac{1}{\sqrt{4\pi\mu_0(t)}} \exp\left(-\frac{(x-Vt)^2}{4\mu_0(t)}\right), \qquad \mu_0(t) = \kappa + \sigma_2^0 t.$$

  The level jump time t_0—at which all the cells at level $\overline{\ell}$ cease to exist—can be estimated by

$$\frac{C_3 2^{-3\overline{\ell}}}{\mu_0(t_0)^2} = \epsilon, \qquad \text{hence} \qquad t_0 = \frac{1}{\sigma_2^0}\left(\frac{\sqrt{C_3}}{\epsilon^{1/2} 2^{3\overline{\ell}/2}} - \kappa\right),$$

  the time at which the $W^{3,\infty}$ semi-norm of the exact solution of the modified equation falls below the threshold ϵ. The fact that $t_0 > 0$ corresponds to the fact that the choice of κ and ϵ were such that the finest level $\overline{\ell}$ was reached when multiresolution is used on the initial datum. The solution at the jump time t_0 shall be the initial datum of another heat equation with a different diffusion coefficient.

- $\Delta\ell = 1$. Again, for $t > t_0$, we have, if we were on a uniform mesh

$$u(t,x) = \frac{1}{\sqrt{4\pi\mu_1(t)}} \exp\left(-\frac{(x-Vt)^2}{4\mu_1(t)}\right), \qquad \mu_1(t) = \mu_0(t_0) + \sigma_2^1 (t - t_0).$$

The level jump time t_1 can be estimated by

$$\frac{C_3 2^{-3(\bar{\ell}-1)}}{\mu_1(t_1)^2} = \frac{\epsilon}{2}, \qquad \text{hence} \qquad t_1 = t_0 + \frac{1}{\sigma_2^1}\left(\frac{\sqrt{C_3}}{(\epsilon/2)^{1/2}2^{3(\bar{\ell}-1)/2}} - \mu_0(t_0)\right).$$

Hence for general $\Delta\ell \geq 1$, we have

$$\mu_{\Delta\ell}(t) = \mu_{\Delta\ell-1}(t_{\Delta\ell-1}) + \sigma_2^{\Delta\ell}(t - t_{\Delta\ell-1}), \qquad t_{\Delta\ell} = t_{\Delta\ell-1} + \frac{1}{\sigma_2^{\Delta\ell}}\left(\frac{\sqrt{C_3}}{(\epsilon/2^{\Delta\ell})^{1/2}2^{3(\bar{\ell}-\Delta\ell)/2}} - \mu_{\Delta\ell-1}(t_{\Delta\ell-1})\right).$$

Otherwise said

$$t_{\Delta\ell} = t_{\Delta\ell-1} + \frac{2^{\bar{\ell}+1}}{V(2^{\Delta\ell} - V/\lambda)}\left(\frac{\sqrt{C_3}2^{2\Delta\ell}}{\epsilon^{1/2}2^{3\bar{\ell}/2}} - \mu_{\Delta\ell-1}(t_{\Delta\ell-1})\right).$$

Table 4.11: Jump times for the upwind scheme using $\hat{\gamma} = 0$.

$\Delta\ell$	Theoretical $t_{\Delta\ell}$	Measured $t_{\Delta\ell}$	Relative error
1	1.73858	1.67041	-3,92%
2	5.52517	4.4375	-19,69 %
3	12.0165	10.8066	-10,07 %

We consider $\kappa = 5\text{e}-4$, $V = 1$, $\lambda = 2$, $\epsilon = 1.5\text{e}-4$ for the simulation. The results are given in Table 4.11, showing good agreement between the theoretical jump times computed using the modified equations and the one by the numerical simulation. This shows that the actual behavior of the numerical solution is well represented by the modified equation.

4.7 Conclusions of Chapter 4

In Chapter 4, we have briefly recalled of adaptive Finite Volume schemes based on multiresolution are constructed. Then, we have performed a modified equation analysis—analogously to Section 3.1—taking advantage of the "reconstruction flattening" both at the level of the leaves in the adaptive tree and on the finest level. This shows that the perturbations with respect to the reference scheme start from order $2\hat{\gamma}+2$ in the space step Δx, where $\hat{\gamma}$ is the number of neighbors considered by the prediction operator used for the computation of the fluxes in the adaptive numerical method. This result is integrated into the error analysis besides the classical estimate as function of the threshold ϵ. Numerical examples show that the modified equations—despite the difficulties linked with the fact that we have to introduce rough upper bounds on the effect of the moving adaptive mesh—provide an additional insight into the behavior of the adaptive multiresolution Finite Volume schemes.

Summary and perspectives of Part I

In Part I, we have first introduced a mesh adaptation strategy and a way of utilizing lattice Boltzmann schemes on adaptive grids, see Chapter 2. The proposed strategy meets the requirements that we have defined at the very beginning, that is, it ensures error control; it evolves the mesh dynamically in time; it reduces the memory footprint of the schemes and finally it is independent of the specific problem at hand. Additional important numerical properties of the proposed strategy—going beyond the requested constraints—have been investigated in Chapter 3. It turns out that our approach ensures that the adaptive scheme behaves like the original scheme at high order in the discretization parameter and that this ensures to strongly reduce wave reflections at mesh jumps, which are a stumbling block for numerical methods on adaptive meshes. Finally, in Chapter 4, we have adapted the tool used for the analysis of the adaptive lattice Boltzmann schemes to study adaptive Finite Volume schemes.

An important topic which has been left from the discussion of Part I concerns the actual implementation of the proposed numerical strategy. The objective is to implement the methods of Part I in a wider and more general library which could be reused for other applications. This is the topic covered in Part II.

Part II

Data structure and implementation

General introduction

In Part I, we have introduced and tested novel adaptive lattice Boltzmann schemes based on multiresolution. However, very little has been said about the actual implementation to conduct the numerical simulations. Still, a well-designed and structured code is of the foremost importance both to achieve a real speed-up in the computations and to be able to easily implement new and general numerical schemes.

Aim and structure of Part II

The aim of Part II is to provide the main traits of the implementation used to perform the numerical tests in Part I. To this end, Chapter 5 describes the core data structure and the operations to represent and handle Cartesian adaptive grids, which yield the code SAMURAI[1] (**S**tructured **A**daptive Mesh and **Mu**lti**r**esolution based on **A**lgebra of **I**ntervals). Once this general framework—which will be useful for our team and collaborators beyond the scope of this thesis—we showcase how we "specialize" SAMURAI—see Chapter 6—to implement mesh adaptation by multiresolution and adaptive lattice Boltzmann schemes as presented in Chapter 2.

Collaborations

The code SAMURAI is the result of a joint collaboration with H. Leclerc, who wrote a first implementation of the data structure SAMURAI relies on. This work has been carried on by L. Gouarin—now principal developer of SAMURAI—with whom I have strongly interacted during the whole duration of my thesis and to whom I credit a fundamental role in the material presented in Part II.

[1] Can be found on: https://github.com/hpc-maths/samurai

Chapter 5

SAMURAI: a general interval-based data structure

General context and motivation

Numerical simulations of real phenomena often feature areas where the numerical solution presents intricate dynamics and rapid variations and thus need to be carefully followed by a fine discretization to capture the actual physics of the problem. This calls for mesh adaptation, aiming at reducing the computational cost and memory occupation where the solution is smooth, still allowing for massive refinement where the solution shows concerning and complex behavior.

State of the art

Many codes for performing mesh adaptation with different objectives and design criteria exist. It is thus difficult to provide a general and complete overview of all the available approaches. One code, allowing to adapt meshes using AMR with a patch-based standpoint, is SAMRAI [Wissink, 2001]. Another code to handle patch-based AMR is AMReX [Zhang et al., 2021]. The successful code `p4est` [Burstedde et al., 2011] allows to use cell-based AMR in the context of forests of quad/oc-trees, using a space-filling curve (or z-curve) for indexing purpose. Codes like canoP [Drui, 2017, Wargnier, 2019] rely on `p4est` for mesh adaptation and allow to implement Finite Volume schemes for various applications benefitting from the speed-up secured by mesh adaptation. As far as adaptation using multiresolution is concerned, one can refer to the thorough discussion of efficient implementations and parallelizations by [Brix et al., 2011] and to the codes MBARETE [Duarte, 2011, Descombes et al., 2017] or WABBIT [Engels et al., 2021].

Aims and structure of Chapter 5

Considering the heterogeneity and the specialization of the codes available in the literature, the purpose of Chapter 5 is to present SAMURAI, which aims at being a framework—written essentially using the C++ language—for handling adaptive meshes beyond the scope of this thesis, featuring codes to be reused for other applications. Chapter 5 is structured as follows. In Section 5.1, we state the design principles of SAMURAI to make it suitable to the vastest possible range of applications. Complying with these principles, the core of SAMURAI—namely intervals of relative integers—are discussed in Section 5.2. The storage of data on adaptive mesh is rapidly tackled in Section 5.3. The algebra of set and the corresponding operators between sets of cell are another important part of SAMURAI, presented in Section 5.4. Efficiency tests on the data structure are presented in Section 5.5. The conclusions and perspectives are drawn in Section 5.6.

Contents

5.1 Criteria and design principles

We start by defining the essential criteria guiding us in the definition of the data structure and the library built upon and used to handle multi-level Cartesian grids. These are:

1. **Generality and flexibility**, being capable of handling both grids made up of points and control volumes. Moreover, the structure has to be suitable to implement various mesh adaptation strategies, such as patch-based AMR, cell-based AMR and multiresolution based both on points and volumes.
2. **Maximize data contiguity** for the stored values. This is of paramount importance when numerical schemes have to access data.
3. **Ease the writing of operators**, both used to perform mesh adaptation/related operations and to implement numerical schemes.

More in detail, the previous constraints inspire a certain number of design principles.

- Compress the mesh according to the level-wise spatial connectivity along each Cartesian axis. Practically, this signifies that large patches of contiguous cells/points at the same level of resolution shall be gathered as far as their representation in memory is concerned.
- Achieve fast look-up for a cell into the structure, especially for parents and neighbors. This is particularly useful when utilizing numerical schemes on the hybrid mesh.
- Maximize the memory contiguity of the stored data to allow for caching and vectorization, contrarily to the approach by the z-curve.
- Facilitate inter-level operations which are common in many numerical techniques.
- Allow for a time evolution of the hybrid mesh efficiently.
- Give the possibility of writing numerical schemes in a transparent way as if one were on a uniform mesh.

5.2 Interval-based data structure

The major point towards achieving an efficient representation of the connectivity and to compress the grid according to this feature is to consider blocks of contiguous cells along each axis, grouping them level-by-level. The data structure to represent a hybrid partition of the domain Ω—namely the complete leaves $S(\Lambda)$ (*cf.* Section 2.3.3)—or any other set of cells (in the whole Chapter 5, we shall use Figure 5.1 as example) can be summarized as, for $d = 3$

- a list for each level $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$,
- containing a list of intervals along the z direction,
- pointing to lists of intervals along the y direction,
- pointing to lists of intervals along the x direction.

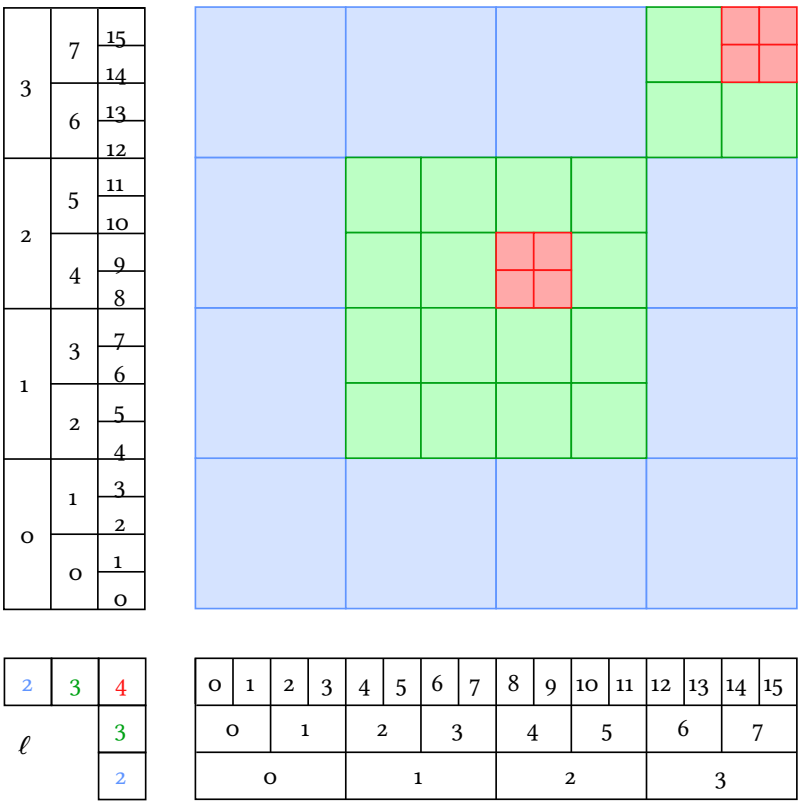


Figure 5.1: Example of hybrid set of cells, in particular $S(\Lambda)$, for $d = 2$.

It is worthwhile observing that thanks to the nested structure of this implementation as far as Cartesian directions are concerned, it can be extended to an arbitrary number d of axes. Moreover, the first Cartesian axis—which is the x axis and is always present regardless of the dimension d—constitutes the end-point of every recursive procedure. Still, this is a choice that we adopted and different ways of operating are equally possible.

The whole construction has been deeply inspired by one of the most successful representations for sparse matrices, namely the CSR (Compressed Sparse Row) format [Tinney and Walker, 1967, Davis, 2006]. We shall try to compare our approach with this one as much as possible.

5.2.1 Intervals

The basic data structure in our implementation is the one-dimensional interval of relative integers $\mathbb{Z}$—which shall be used along each axis x, y and z—under the form

$$I = [\![k_{\text{start}}, k_{\text{end}}[\![\ @ \ \text{offset} \ : \ \text{step}. \tag{5.1}$$

An interval I is defined by its start value $k_{\text{start}} \in \mathbb{Z}$ and end value $k_{\text{end}} \in \mathbb{Z}$ (not included). These attributes are going to vary in the authorized set of values according to the considered level of resolution ℓ. Two additional attributes are considered, namely

- the offset $\in \mathbb{Z}$, which role is going to be clarified soon and differs according to the fact that the considered interval pertains to the x direction or the y and z ones;
- the step $\in \mathbb{N}^*$, which represents the length of the steps used when navigating through the cells of the interval I.

Observe that—in our implementation—the level of resolution ℓ is not stored in the interval I itself but is determined by the structure gathering them together level-by-level. For any interval I given by (5.1), we will indicate

$$\text{start}(I) = k_{\text{start}}, \qquad \text{end}(I) = k_{\text{end}}.$$

Let us stress once more that the indices of an interval cannot make sense when detached from the corresponding level of resolution.

5.2.2 `CellList`: condensation along the x axis

We here present how to compress the representation of a level-heterogeneous set of cells—such as $S(\Lambda)$—level-by-level, by recognizing contiguous cells along the x axis, which are going to form a unique interval. At the end, the structure `CellList` is nothing else than a series of maps, one for each level, containing a list of intervals which have been "condensed" according to the spatial connectivity along x. In Section 5.2.2 and Algorithm 1, we forget about offset and step since they do not play any role in the procedure. The process to achieve such result is presented in Algorithm 1, which takes a set of cells, for example $S(\Lambda)$, as input and returns as much maps as the number of levels, each one using tuples of size $d-1$ as keys and lists of intervals as values. Though we present it for $d=3$, thanks to the intrinsic independence of the algorithm from the size $d-1$ of the key of the maps, it naturally works for any spatial dimension d, with the notable case of a unique null key for $d=1$.

Algorithm 1 Condensation along x

Input: $S(\Lambda)$.
for $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ **do** ▷ Navigate through levels
  $\mathrm{Map}_\ell = \{\,\}$.
  for $k_z \in$ admissible range **do** ▷ Navigate through admissible z-coordinates
    for $k_y \in$ admissible range **do** ▷ Navigate through admissible y-coordinates
      $L_{\text{fill}} = (\,)$. ▷ List of intervals to fill at these k_y and k_z
      $I_{\text{curr}} = \varnothing$. ▷ Current interval to build at this k_y and k_z
      for $k_x \in$ admissible range with $(\ell, \boldsymbol{k}) \in S(\Lambda)$ **do** ▷ Navigate through $S(\Lambda)$ following the x-axis
        if $I_{\text{curr}} \equiv \varnothing$ **then** ▷ Starting to fill at these k_y and k_z
          $I_{\text{curr}} = [\![k_x, k_x+1[\![$. ▷ The current interval is this cell
        else ▷ We have already filled something at these k_y and k_z
          if $k_x \equiv \mathrm{end}(I_{\text{curr}})$ **then** ▷ New contiguous cell along x
            $\mathrm{end}(I_{\text{curr}}) = \mathrm{end}(I_{\text{curr}}) + 1$. ▷ Add this cell to the current interval
          else ▷ The contiguity chain along x is broken
            Append I_{curr} to L_{fill}. ▷ Store the old interval
            $I_{\text{curr}} = [\![k_x, k_x+1[\![$. ▷ The current interval is this cell
          end if
        end if
      end for
      Append I_{curr} to L_{fill}. ▷ Store the old interval
      Add L_{fill} to $\mathrm{Map}_\ell[(k_y, k_z)]$. ▷ Finished to scan for these k_y and k_z
    end for
  end for
end for
Output: $\mathrm{Map}_{\underline{\ell}}, \ldots, \mathrm{Map}_{\overline{\ell}}$.

In order to practically provide an example of the computation carried by Algorithm 1, we apply the algorithm to set of cells presented on Figure 5.1. The outcome is illustrated in Figure 5.2 and also reads:

$$
\begin{aligned}
\mathrm{Map}_2[(0)] &= ([\![0,4[\![), & \mathrm{Map}_2[(1)] &= ([\![0,1[\![, [\![3,4[\![),\\
\mathrm{Map}_2[(2)] &= ([\![0,1[\![, [\![3,4[\![), & \mathrm{Map}_2[(3)] &= ([\![0,4[\![),\\
\mathrm{Map}_3[(2)] &= ([\![2,6[\![), & \mathrm{Map}_3[(3)] &= ([\![2,6[\![),\\
\mathrm{Map}_3[(4)] &= ([\![2,4[\![, [\![5,6[\![), & \mathrm{Map}_3[(5)] &= ([\![2,6[\![),\\
\mathrm{Map}_3[(6)] &= ([\![6,8[\![), & \mathrm{Map}_3[(7)] &= ([\![6,7[\![),\\
\mathrm{Map}_4[(8)] &= ([\![8,10[\![), & \mathrm{Map}_4[(9)] &= ([\![8,10[\![),\\
\mathrm{Map}_4[(14)] &= ([\![14,16[\![), & \mathrm{Map}_4[(15)] &= ([\![14,16[\![).
\end{aligned}
$$

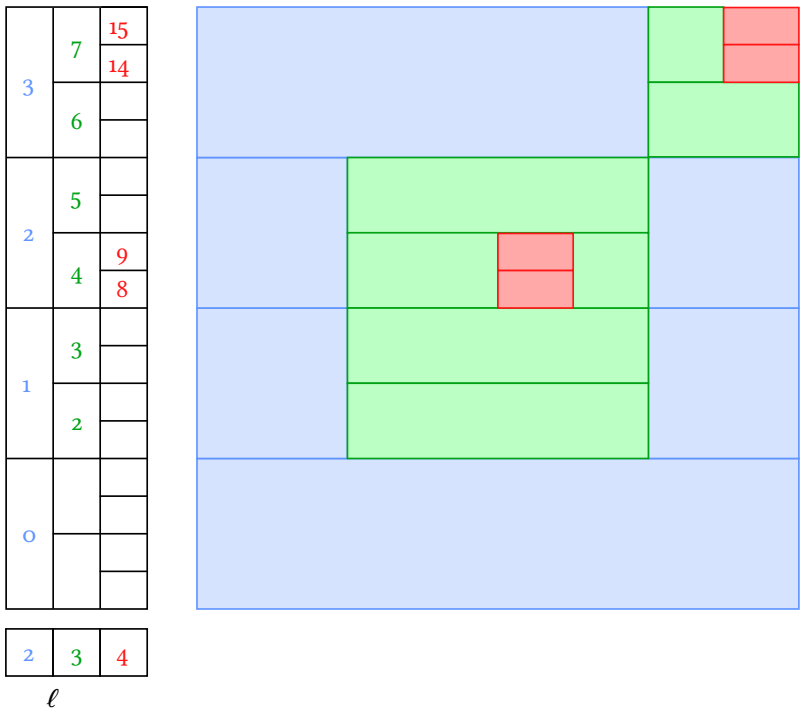


Figure 5.2: Example of reduction to intervals along the x axis on the example of Figure 5.1 using Algorithm 1.

To make the link with the CSR format for sparse matrices, the condensed intervals along x are the elements of the column indices vector where contiguous columns with non-zero entries at the same row are merged together. In this type of storage, one also needs the vector of row indices: the construction of its analogous for Cartesian meshes as well as the condensation of contiguous cell along the remaining axis is the topic of the following Section 5.2.3.

5.2.3 `CellArray`: condensation along the other axes

We now explain how the condensation based on contiguity is done along the other axes—which are present whenever $d > 1$. This is the exact analogous of the need for a row indices vector in the CSR storage for sparse matrices. This yields a structure called `CellArray`, or compressed multi-axes representation. The issue is solved using the offset of each interval pertaining to y, z, *etc.* as well as a series of vectors

$$L^x_\ell\text{-ptr}, \quad L^y_\ell\text{-ptr}, \quad \ldots,$$

for $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$, being the analogous of the row indices vectors for matrices in the CSR storage format. This allows, at the end of the day, to recompose the multidimensional geometry of the set of cells.

The way of proceeding is detailed in Algorithm 2, where it is interesting to observe that the size of each vector L^x_ℓ-ptr the number of keys in Map_ℓ plus one. In the CSR storage, the size of the row indices vector is $M+1$ where M is the number of rows of the considered matrix. Therefore, we see that there is a correspondence between each element of Map_ℓ for the Cartesian grid and a row of a matrix.

At the end of the process, always taking the examples of the complete leaves $S(\Lambda)$, this set of cells have been transformed to an equivalent compact representation as follows:

$$S(\Lambda) \subset \{(\ell, \boldsymbol{k}) \,:\, \ell \in [\![\underline{\ell}, \overline{\ell}]\!]\} \quad \leftrightarrows \quad \begin{cases} x \text{ axis} & : & L^x_\ell & & & \\ y \text{ axis} & : & L^y_\ell & \text{and} & L^x_\ell\text{-ptr} & \\ & \vdots & & & & \\ & \vdots & & & & \end{cases} \quad \text{for} \quad \ell [\![\underline{\ell}, \overline{\ell}]\!]. \tag{5.2}$$

Algorithm 2 Condensation along the remaining axes

Input: $\mathrm{Map}_{\underline{\ell}}, \ldots, \mathrm{Map}_{\overline{\ell}}$. ▷ We start from the output of Algorithm 1
for $\ell \in [\![\underline{\ell}, \overline{\ell}]\!]$ **do** ▷ Navigate through levels
  $L^y_\ell = (\,)$. ▷ The list of intervals in the y starts empty
  if $\mathrm{Map}_\ell \not\equiv \emptyset$ **then** ▷ If there are cells in the x direction for the level
    $I_{\mathrm{curr}} = \emptyset$. ▷ Current interval to construct along y
    L^x_ℓ-ptr $= \underbrace{(0, \ldots, \ldots)}_{\#(\mathrm{Map}_\ell)+1 \text{ elements}}$.
    $r = 1$.
    for $k_y \in \mathrm{Map}_\ell$ **do** ▷ Navigating through the keys of Map_ℓ
      if $I_{\mathrm{curr}} \equiv \emptyset$ **then** ▷ Starting to fill an interval
        $I_{\mathrm{curr}} = [\![k_y, k_y + 1[\![\;@\; L^x_\ell\text{-ptr}[r] - k_y$.
      else ▷ The interval is already formed
        if $k_y \equiv \mathrm{end}(I_{\mathrm{curr}})$ **then** ▷ New contiguous cell along y
          $\mathrm{end}(I_{\mathrm{curr}}) = \mathrm{end}(I_{\mathrm{curr}}) + 1$. ▷ Add this cell to the current interval
        else ▷ There is no contiguity in y
          Append I_{curr} to L^y_ℓ. ▷ Store the old interval
          $I_{\mathrm{curr}} = [\![k_y, k_y + 1[\![\;@\; L^x_\ell\text{-ptr}[r] - k_y$. ▷ The current interval is a cell
        end if
      end if
      $L^x_\ell\text{-ptr}[r+1] = L^x_\ell\text{-ptr}[r] + \#$ of x intervals in $\mathrm{Map}_\ell[(k_y)]$. ▷ Update the vector of offsets
      $r = r + 1$.
    end for
    Append I_{curr} to L^y_ℓ.
  end if
end for
Output: $L^y_{\underline{\ell}}, \ldots, L^y_{\overline{\ell}}$ and $L^x_{\underline{\ell}}\text{-ptr}, \ldots, L^x_{\overline{\ell}}\text{-ptr}$.

Here, each list of intervals along the x axis L^x_ℓ has been simply created from Map_ℓ. The list of intervals along the other axes are denoted by L^y_ℓ, ... For the y-axis, we write L^x_ℓ-ptr to indicate that it is used to point towards intervals along x, namely in L^x_ℓ.

5.2.4 Role of the offset

We now explain more in detail the role of the offset, which is two-fold and depends on the fact that the associated interval pertains the x-axis or one of the remaining axes y, z, ...

- In the first case, offset is used to recover the index of the stored data for the given interval. They are stored in a linear structure according to the principle of utilizing contiguous memory emplacements for cells belonging to the same interval and for intervals belonging to the same level of resolution. The idea behind this numbering strategy is illustrated in Figure 5.3. We start numbering from the coarsest level $\underline{\ell}$ up to finest level $\overline{\ell}$, navigating for fixed y and z indices in the direction of x. Then, the offset is constructed in the following way: for the cells of an interval $I = [\![k_{\mathrm{start}}, k_{\mathrm{end}}[\![\;@\; \mathrm{offset} : \mathrm{step}$, the positions of the corresponding fields in the linear memory structure are in the interval $[\![k_{\mathrm{start}} + \mathrm{offset}, k_{\mathrm{end}} + \mathrm{offset}[\![$.
- In the second case, the offsets are used to construct "pointers" between list of intervals on different Cartesian axes (link x with y, y with z, *etc.*) as done by Algorithm 2. Given an interval $I \in L^y_\ell$, the corresponding offset stored with I gives, for every cell making up the interval $k_y \in [\![\mathrm{start}(I), \mathrm{end}(I)]\!]$, the indices $r_{k_y} = k_y + \mathrm{offset}$. Therefore, the intervals along the x-axis crossing those in I on the y-axis are given by the elements in L^x_ℓ indexed between $L^x_\ell\text{-ptr}[r_{k_y}]$ and $L^x_\ell\text{-ptr}[r_{k_y}+1]$, the latter not included.

The illustrate on Figure 5.2, we obtain, for levels $\ell = 2$ and $\ell = 4$

$$L^x_2 = ([\![0,4[\![, [\![0,1[\![, [\![3,4[\![, [\![0,1[\![, [\![3,4[\![, [\![0,3[\![), \qquad L^y_2 = ([\![0,4[\![\;@\; 0), \qquad L^x_2\text{-ptr} = (0,1,3,5,6),$$
$$\ldots$$

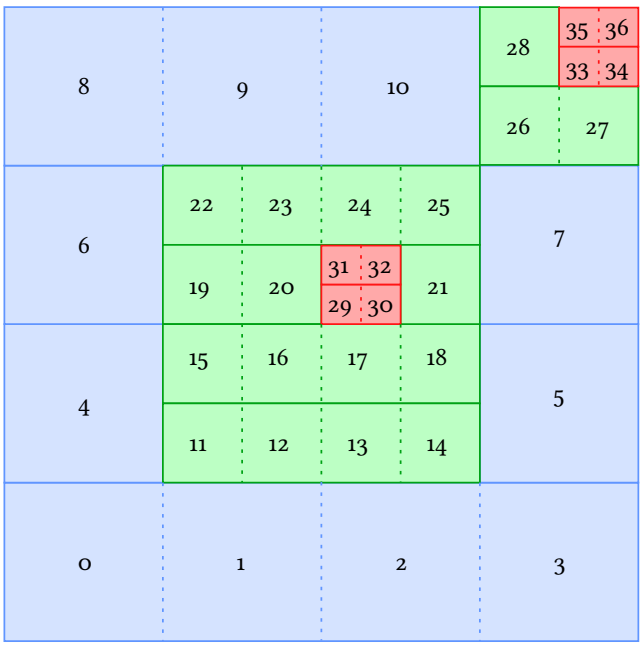


Figure 5.3: Example of indices for the data defined on each cell from Figure 5.1 in the linear structure used for storage. We start assigning from the coarsest level $\underline{\ell}$ to the finest level $\overline{\ell}$ and navigating along the x direction.

$$L_4^x = (\llbracket 8,10\llbracket, \llbracket 8,10\llbracket, \llbracket 14,16\llbracket, \llbracket 14,16\llbracket), \qquad L_4^y = (\llbracket 8,10\llbracket \ @\ -8, \llbracket 14,16\llbracket \ @\ -12), \qquad L_4^x\text{-ptr} = (0,1,3,5,6).$$

5.3 xtensor: A LINEAR DATA STRUCTURE FOR STORAGE

In Section 5.2.4, we have explained how the intervals are linked to the indices in the linear structure used to store fields defined of sets of cells. However, we have not specified the specific choice of linear structure that we use. We employ the C++ library `xtensor`, in order to take advantage of lazy evaluations of expressions and to easily write code as we were using the `numpy` package in `Python`. This will allow to access a given field simply by specifying the level of the corresponding cell ℓ as well as its indices $\boldsymbol{k}$ without having to care about the actual access in memory. For example, when $d = 2$

```
field(level, idx_x, idx_y) = xt::pow(field(level, idx_x, idx_y), 2);
```

takes the field at level `level` with indices `idx_x` (this can be a whole interval of cells) and `idx_y` and raises it to square, element-by-element, with lazy evaluation.

5.4 Algebra of sets

Apart from the compression achieved by considering intervals of contiguous cells, another important advantage of the representation introduced in Section 5.2 is that it allows to perform geometrical operations between groups of cells, even between different levels, as operations on the corresponding sets of intervals. These operations can be used—for example—to select certain groups of cells (*e.g.* ghost cells, neighbors, parents, cells closed to the boundary, cells next to a level jump, *etc.*) to perform modifications on the meshes and access or modify the data stored on them. To this end, SAMURAI features what we call an "algebra of sets". To justify this even more—from the perspective of mesh compression—think about an adaptive mesh made up of patches of cells where computations are performed, surrounded by ghost cells which are employed to perform data exchanges between levels. The spatial contiguity of ghost cells alone is weak, but very good compression factors, thanks to the contiguity, can be achieved by gathering them with the cells. Still, one would be able to quickly and simply

recover the ghost cells alone: this is possible with the algebra of sets.

In order to pass from sets of intervals to actual cells, we introduce, for a given set of indices $\Lambda \subset \{(\ell, \boldsymbol{k}) \ : \ \ell \in [\![\underline{\ell}, \overline{\ell}]\!]\}$, the corresponding cells $\mathcal{C}(\Lambda)$ given by:

$$\mathcal{C}(\Lambda) := \bigcup_{(\ell,\boldsymbol{k})\in\Lambda} C_{\ell,\boldsymbol{k}},$$

where the cells $C_{\ell,\boldsymbol{k}}$ have been defined in (2.8). It should be kept in mind that Λ is stored as represented in the right part of (5.2). For example, if Λ represents a tree, then $\mathcal{C}(S(\Lambda)) = \Omega$, that is, the complete leaves of the tree pave the whole domain Ω. We present the operations using volumes, but they can be employed with points by charging them with Dirac measures on points instead of on volumes.

5.4.1 Operators

We introduce the most important operators that we have implemented in SAMURAI. It is possible and easy to create new operators according to the user's need. To illustrate them—since they are often binary operators—consider two sets of cells

$$\Lambda_1 \subset \{(\ell_1, \boldsymbol{k})\}, \qquad \Lambda_2 \subset \{(\ell_2, \boldsymbol{k})\},$$

respectively at level $\ell_1 \in [\![\underline{\ell}, \overline{\ell}]\!]$ and $\ell_2 \in [\![\underline{\ell}, \overline{\ell}]\!]$, not necessarily equal. We introduce:

- The **projection** of the set Λ_1 on a target level ℓ_2, that we indicate by $\mathsf{P}_{\ell_2}(\Lambda_1)$. This operation yields the set of dyadic cells at level ℓ_2 with minimal d-dimensional Lebesgue measure containing the volume $\mathcal{C}(\Lambda_1)$ corresponding to the set Λ_1. This can be written as:

  $$\mathsf{P}_{\ell_2}(\Lambda_1) = \operatorname{argmin}\{|\mathcal{C}(\Lambda)|_d \quad : \quad \Lambda \subset \{(\ell_2, \boldsymbol{k})\} \quad \text{and} \quad \mathcal{C}(\Lambda) \subset \mathcal{C}(\Lambda_1)\}.$$

  As long as $\ell_2 \geq \ell_1$, due to the nesting, the projection presented above is simply given by the formula $\mathsf{P}_{\ell_2}(\Lambda_1) = \{(\ell_2, 2^{\ell_2-\ell_1}\boldsymbol{k}) \ : \ (\ell_1, \boldsymbol{k}) \in \Lambda_1\}$, so that $|\mathcal{C}(\mathsf{P}_{\ell_2}(\Lambda_1))|_d = |\mathcal{C}(\Lambda_1)|_d$. Otherwise, in the general context, $|\mathcal{C}(\mathsf{P}_{\ell_2}(\Lambda_1))|_d \geq |\mathcal{C}(\Lambda_1)|_d$: the Lebesgue measure of the projected cells is larger than the one of the original cells. In the C++ implementation, the projection operator is encoded by `.on(level_2)`, where `level_2` stands for ℓ_2.

- The **union** $\Lambda_1 \sqcup \Lambda_2$ of two sets Λ_1 and Λ_2, given by

  $$\Lambda_1 \sqcup \Lambda_2 = \{(\max(\ell_1, \ell_2), \boldsymbol{k}) \quad : \quad C_{\ell,\boldsymbol{k}} \in \mathcal{C}(\mathsf{P}_{\max(\ell_1,\ell_2)}(\Lambda_1)) \cup \mathcal{C}(\mathsf{P}_{\max(\ell_1,\ell_2)}(\Lambda_2))\}.$$

  This represents the cells belonging either to the projection of Λ_1 or Λ_2 on the level $\max(\ell_1, \ell_2)$. Since the operator involves a projection, we have that $|\mathcal{C}(\Lambda_1 \sqcup \Lambda_2)|_d \geq |\mathcal{C}(\Lambda_1)|_d + |\mathcal{C}(\Lambda_2)|_d$ if $|\mathcal{C}(\Lambda_1) \cap \mathcal{C}(\Lambda_2)|_d = 0$. In the C++ implementation, this operation is denoted by `union_(set_1, set_2)`.

- The **intersection** $\Lambda_1 \sqcap \Lambda_2$ of two sets Λ_1 and Λ_2, given by

  $$\Lambda_1 \sqcap \Lambda_2 = \{(\max(\ell_1, \ell_2), \boldsymbol{k}) \quad : \quad C_{\ell,\boldsymbol{k}} \in \mathcal{C}(\mathsf{P}_{\max(\ell_1,\ell_2)}(\Lambda_1)) \cap \mathcal{C}(\mathsf{P}_{\max(\ell_1,\ell_2)}(\Lambda_2))\}.$$

  This is made up of cells belonging both to the projection of Λ_1 and Λ_2 on the level $\max(\ell_1, \ell_2)$. In the C++ implementation, this operation is denoted by `intersection(set_1, set_2)`.

- The **difference** $\Lambda_1 \rhd \Lambda_2$ of two sets Λ_1 and Λ_2, given by

  $$\Lambda_1 \rhd \Lambda_2 = \{(\max(\ell_1, \ell_2), \boldsymbol{k}) \quad : \quad C_{\ell,\boldsymbol{k}} \in \mathcal{C}(\mathsf{P}_{\max(\ell_1,\ell_2)}(\Lambda_1)) \setminus \mathcal{C}(\mathsf{P}_{\max(\ell_1,\ell_2)}(\Lambda_2))\}.$$

  This yields the cells belonging to the projection of Λ_1 but not to that of Λ_2 on the level $\max(\ell_1, \ell_2)$. In the C++ implementation, this operation is denoted by `difference(set_1, set_2)`.

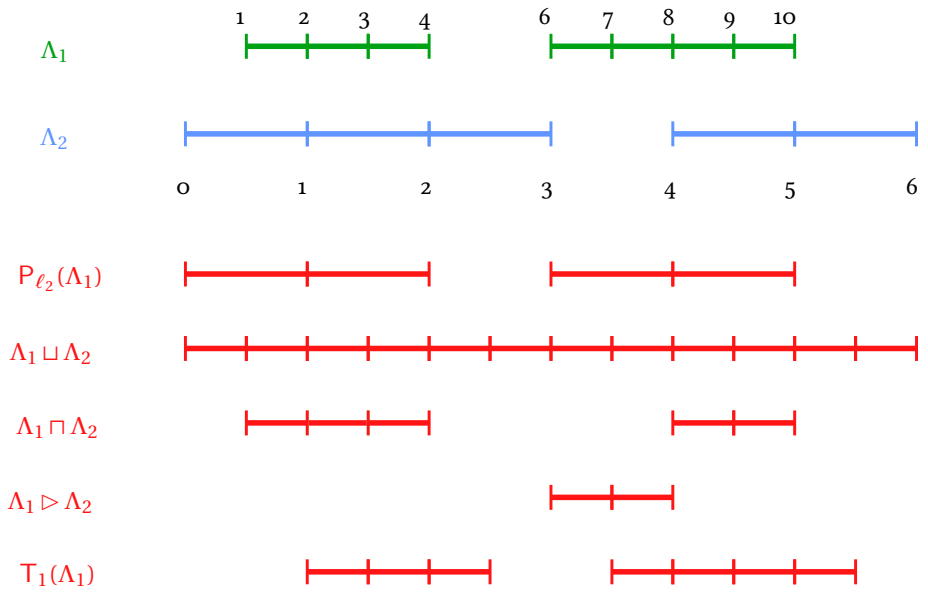


Figure 5.4: Example of operators on two sets Λ_1 (green) and Λ_2 (blue) for $d = 1$. The results of the operations are provided in red.

- The **translation** $T_{\boldsymbol{s}}(\Lambda_1)$ of the set Λ_1 by a given integer stencil $\boldsymbol{s} \in \mathbb{Z}^d$, given by

$$T_{\boldsymbol{s}}(\Lambda_1) = \{(\ell_1, \boldsymbol{k} + \boldsymbol{s}) \quad : \quad (\ell_1, \boldsymbol{k}) \in \Lambda_1\}.$$

  In the C++ implementation, this operator can be called using `translate(set, stencil)`.

An example of the way of acting of these operations is given in Figure 5.4 for the 1D case $d = 1$. Here, one can notice what we have pointed out concerning the Lebesgue measure of the outcomes of these operators.

5.4.2 Implementation

We explain on an example how these operators are practically implemented. We consider the intersection of two sets for $d = 1$, which are written as lists of intervals Λ_1 and Λ_2. This is present in Algorithm 3.

5.5 Performance tests

The test we present here aims at evaluating the compression rate obtained using the approach of SAMURAI compared to `p4est` [Burstedde et al., 2011]. To this end, we create a hybrid cover of the square $\Omega = [0, 1]^2$ obtained in the following way. We start with a mesh where the upper-right and lower-left sectors of Ω correspond to one cell each at level $\ell = 1$, whereas the upper-left and lower-right parts contain four cells each at level $\ell = 2$. Then, we iteratively refine $\overline{\ell} - 2$ times whenever the lower corner of a cell has x coordinate smaller than $1/4$ or both the x and y coordinates are equal to $3/4$. An example for $\overline{\ell} = 9$ is given in Figure 5.5.

We measure the size occupied by the representation of such mesh between SAMURAI and `p4est`, see Figure 5.6. Here, `cells` are the leaves of the underlying tree which are the actual cells of the mesh, whereas `all_cells` also include auxiliary cells like ghosts, *etc.*, see Chapter 6. For SAMURAI, we observe that increasing $\overline{\ell}$, thus the number of cells to store (proportionally to $2^{d\overline{\ell}}$), the memory footprint *per* cell decreases, thanks to the storage relying on spatial connectivity. Indeed, the mesh occupies memory proportionally to $2^{(d-1)\overline{\ell}}$. Quite the opposite, for `p4est`, since such rationale is not present in the design of the data structure, the cost per cell remains constant and the occupation rate trends like $2^{d\overline{\ell}}$. For the very same reason, SAMURAI achieves increasingly better compression rates compared to `p4est` as the size of the problem increases, with memory space taken used by the mesh representation being between 100 and 1000 times smaller than for `p4est`.

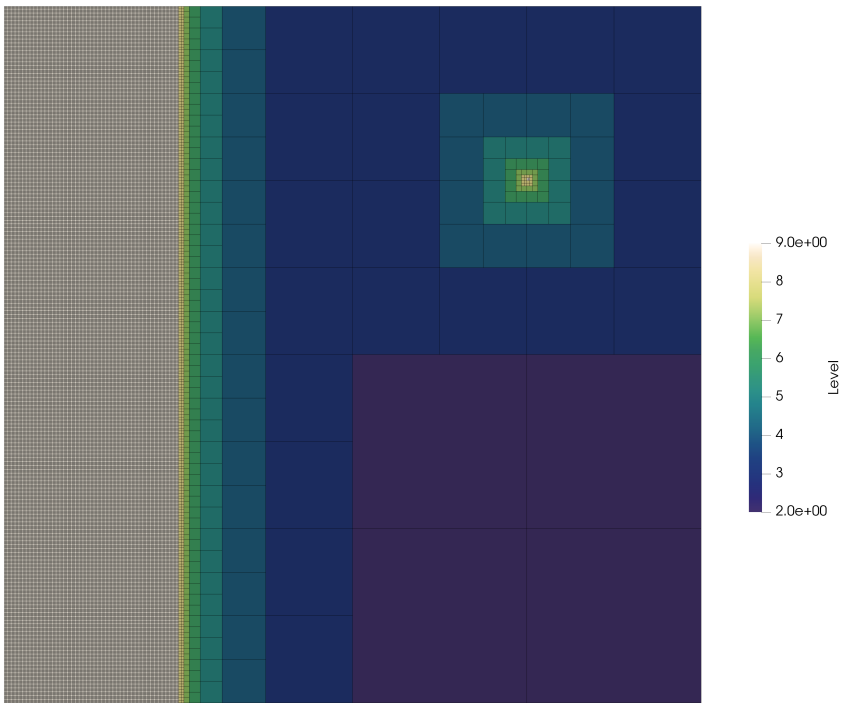


Figure 5.5: Grid used for the comparison of SAMURAI and p4est considering $\overline{\ell} = 9$.

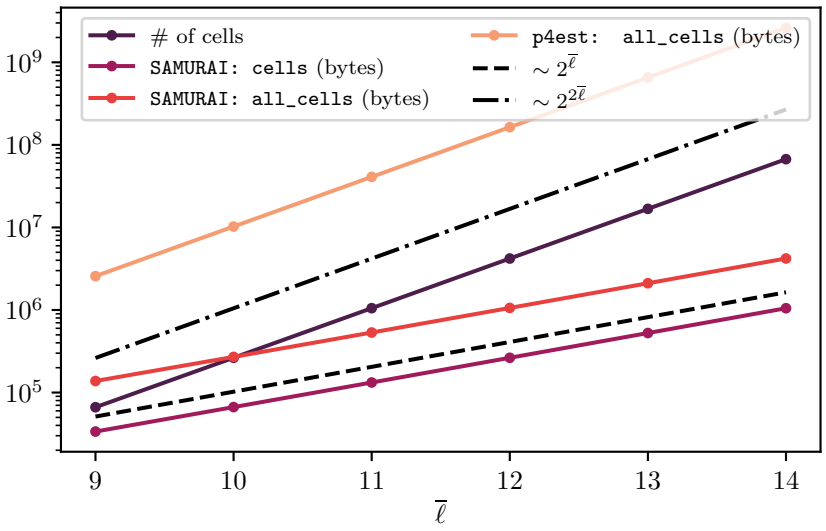


Figure 5.6: Comparison in terms of memory occupation between SAMURAI and p4est

Algorithm 3 Intersection operator implementation

Input: Lists of intervals Λ_1 and Λ_2.
$L = ( )$ ▷ List of intersection intervals
$i_1 = 0$, $k_1 = \text{start}(\Lambda_1[i_1])$. ▷ Indices of the first interval in the list Λ_1
$i_2 = 0$, $k_2 = \text{start}(\Lambda_2[i_2])$. ▷ Indices of the first interval in the list Λ_2
$\text{scan} = \min(k_1, k_2)$. ▷ We start by the smallest indices
$\text{sentinel} = \max(\text{end}(\Lambda_1[-1]), \text{end}(\Lambda_2[-1])) + 1$. ▷ We certainly finish at the largest indices (`numpy` notation)
$I = \emptyset$. ▷ Current interval to fill
inside_interval = false
while scan < sentinel **do**
  if $\text{scan} \in \Lambda_1[i_1]$ and $\text{scan} \in \Lambda_2[i_2]$ and !inside_interval **then** ▷ We are in the intersection and starting
    inside_interval = true ▷ Flag to know that we started a new interval
    $\text{start}(I) = \text{scan}$ ▷ Setting the start of the interval
  end if
  if $\text{scan} \notin \Lambda_1[i_1]$ or $\text{scan} \notin \Lambda_2[i_2]$ and inside_interval **then** ▷ Exiting from an intersection
    inside_interval = false
    $\text{end}(I) = \text{scan}$ ▷ Close the interval
    Append I to L ▷ Add to the output list
    $I = \emptyset$ ▷ We start again with an empty interval
  end if
  $\text{increment}(i_1, k_1, \Lambda_1, \text{sentinel})$ ▷ Implemented in Algorithm 4
  $\text{increment}(i_2, k_2, \Lambda_2, \text{sentinel})$
  $\text{scan} = \min(k_1, k_2)$ ▷ The current indices is the minimum between indices for each set
end while
Output: L.

Algorithm 4 Increment implementation

Input: i, k, Λ and sentinel.
if $\exists \tilde{i}$ such that $k \equiv \text{end}(\Lambda[\tilde{i}])$ **then** ▷ The current indices k is on the end of some interval in the set Λ
  $i = i + 1$ ▷ Go to the next interval
  if $i > \#(\Lambda)$ **then** ▷ Going beyond length
    $\text{sentinel} = k$
  else
    $k = \text{start}(\Lambda[i])$ ▷ We go to the start of the next interval
  end if
end if
if $\exists \tilde{i}$ such that $k \equiv \text{start}(\Lambda[\tilde{i}])$ **then** ▷ The current indices k is on the start of some interval in the set Λ
  $k = \text{end}(\Lambda[i])$ ▷ We go to the end of the current interval
end if

5.6 Conclusions

In Chapter 5, we have introduced a data structure based on integer intervals that allows to represent multi-level Cartesian meshes in a compact fashion. As showcased, our way of compressing the mesh allows for impressive compression rates compared to libraries such as `p4est`. Moreover, we introduced operators between groups of cells which allows to operate on certain elements of the mesh. An issue that deserves attention for future investigations and which is currently under scrutiny concerns the parallelization and the distribution of the data structure. This point is of the foremost importance to take advantage of modern computer architectures. Beyond the implementation of the methods presented in this thesis, see Chapter 6, `SAMURAI` is used in a large panel of applications—which development is still an ongoing work—by collaborators. Applications range from solving linear systems stemming from the Poisson equation to the solution of the incompressible Navier-Stokes system.

Chapter 6

Multiresolution and adaptive lattice Boltzmann schemes implementation

Aims and structure of Chapter 6

The aim of Chapter 6 is to present our implementation—within SAMURAI—of the mesh adaptation based on multiresolution introduced in Section 2.3 and Section 2.4 and concerning the adaptive lattice Boltzmann schemes of Section 2.5, with the optimizations explained in Section 2.5.4.1. To this end, Chapter 6 is structured as follows. In order to avoid encoding the adaptive mesh as a tree structure, we introduce, in Section 6.1, several categories of cells. These are used in Section 6.2 to implement mesh adaptation by multiresolution as presented in Section 2.3 and Section 2.4. Once the adaptive mesh is available, we illustrate—*cf.* Section 6.3—how the adaptive lattice Boltzmann schemes of Section 2.5 are implemented within SAMURAI and we devote particular care to the issues concerning an optimized implementation of the reconstruction operator *via* its flattening (see Section 2.5.4.1). We conclude in Section 6.4. Since—besides technicalities which however do not prevent to understand the main ideas—the implementation has been kept user-friendly and high-level as much as possible, we will complement Chapter 6 with real C++ code.

Contents

6.1 Cell categories and iterative implementation

We have already pointed out the fact that for a given hybrid mesh of the domain, we do not store the associated graded tree. Instead, we use a dedicated data structure based on different categories of cells and an iterative computation of the details (and thus of the multiresolution transform). Hence, mesh adaptation is performed by

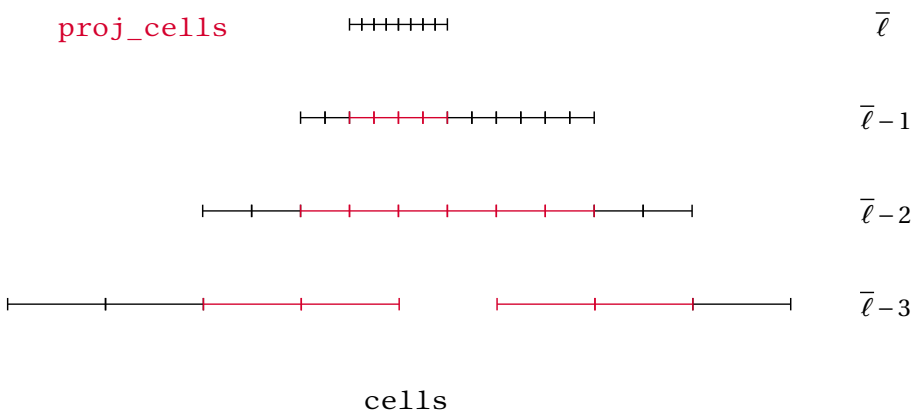


Figure 6.1: Illustration of what the `proj_cells` (in red) are, once the leaves `cells` (in black) are given, for a prediction operator considering one neighbor $\gamma = 1$.

successive passes in an iterative way. The categories that we consider to perform multiresolution are:

$$\begin{cases} \texttt{cells} \\ \texttt{cells\_and\_ghosts} \\ \texttt{proj\_cells} \\ \texttt{all\_cells} \\ \texttt{(overleaves)} \end{cases}$$

where the last category is not used in the adaptation process but uniquely to deploy the numerical scheme. Eventually, this category could be eliminated with a more optimized implementation. For the other ones, `cells` represent the complete leaves $S(\Lambda)$ of the adaptive tree Λ, which make up the hybrid partition of the domain Ω and on which we evolve the numerical solution. `cells_and_ghosts` is obtained by adding ghost cells level-by-level. `proj_cells` is made up of the cells at level $\ell - 1$ which are needed to compute the prediction on the `cells` at level ℓ, excluding `cells` at level $\ell - 1$. This means that it is made up of the projection of `cells` one level below plus the neighbors needed by the prediction operator $\boldsymbol{P}_{\triangle}$, except when they intersect other `cells`. An example is provided in Figure 6.1. `all_cells` gathers all the previous categories together.

6.2 Mesh adaptation

The mesh adaptation process is entirely based on a field called `tag`, which indicates the destiny of each cell inside the mesh. The possible logical values for `tag` are the following

$$\begin{cases} \texttt{keep} & \leftrightarrow & 1, \\ \texttt{coarsen} & \leftrightarrow & 2, \\ \texttt{refine} & \leftrightarrow & 4, \\ \texttt{enlarge} & \leftrightarrow & 8. \end{cases}$$

A cell tagged with `keep` shall be kept as it is in the refinement process. Quite the opposite, if the `coarsen` tag is applied, the cell will be coarsened. When `refine` is used, the cell will be refined. Finally, the `enlarge` tag is used to add neighboring cells, which shall be created at the end of the process. The integers corresponding to each flag are chosen to enforce a certain priority between tags: `refine` has priority over `keep`, which has priority over `coarsen`. This allows to recombine tags in a bit-wise fashion ensuring that we consider the most restrictive condition. The C++ code performing the mesh adaptation as well as the update of the solution field—which we use as "starting point" to explain the code in more detail—reads as follows.

```
template <class TField, class Func>
void Adapt<TField, Func>::operator()(double eps, double regularity)
{
    auto & mesh = m_field.mesh();
    std::size_t min_level = mesh.min_level();
    std::size_t max_level = mesh.max_level();

    mesh_t mesh_old = mesh;
    field_type field_old(m_field.name(), mesh_old);
    field_old.array() = m_field.array();
    for (std::size_t i = 0; i < max_level - min_level; ++i)
    {
        m_detail.resize();
        m_tag.resize();
        m_tag.fill(0);
        if (harten(i, eps, regularity, field_old))
            break;
    }
}
```

Besides the self-explanatory nature of this code, we emphasize the fact that `eps` on Line 16 is the threshold parameter ϵ, whereas `regularity` is nothing but the value of ν where the solution of the problem is assumed to be of regularity $W^{\nu,\infty}$. Then, the function `harten` implemented as follows.

```
bool Adapt<TField, Func>::harten(std::size_t ite, double eps, double regularity, field_type& field_old)
{
    auto & mesh = m_field.mesh();
    std::size_t min_level = mesh.min_level(), max_level = mesh.max_level();
    for_each_cell(mesh[mesh_id_t::cells], [&](auto &cell)
    {
        m_tag[cell] = static_cast<int>(CellFlag::keep);
    });
    // ...
    for (std::size_t level =  ((min_level > 0)? min_level - 1: 0); level < max_level - ite; ++level)
    {
        auto subset = intersection(mesh[mesh_id_t::all_cells][level],
                                   mesh[mesh_id_t::cells][level + 1])
                      .on(level);
        subset.apply_op(compute_detail(m_detail, m_field)); // Implemented in Section 6.2.1
    }

    for (std::size_t level = min_level; level <= max_level - ite; ++level)
    {
        double exponent = dim * (max_level - level);
        double eps_l = std::pow(2., -exponent) * eps;
        double regularity_to_use = std::min(regularity, 3.0) + dim;
        auto subset_1 = intersection(mesh[mesh_id_t::cells][level],
                                     mesh[mesh_id_t::all_cells][level-1])
                        .on(level-1);
        // Coarsening implemented in Section 6.2.2
        subset_1.apply_op(to_coarsen_mr(m_detail, m_tag, eps_l, min_level));
        // Refinement implemented in Section 6.2.3
        subset_1.apply_op(to_refine_mr(m_detail, m_tag, (pow(2.0, regularity_to_use)) * eps_l, max_level));
    }

    for (std::size_t level = min_level; level <= max_level - ite; ++level)
    {
        auto subset_2 = intersection(mesh[mesh_id_t::cells][level],
                                     mesh[mesh_id_t::cells][level]);
        auto subset_3 = intersection(mesh[mesh_id_t::cells_and_ghosts][level],
                                     mesh[mesh_id_t::cells_and_ghosts][level]);
        subset_2.apply_op(enlarge(m_tag)); // Refinement implemented in Section 6.2.3
        subset_2.apply_op(keep_around_refine(m_tag));
        subset_3.apply_op(tag_to_keep<0>(m_tag, CellFlag::enlarge));
    }

    // Coarsening grading procedure, implemented in Section 6.2.4
    for (std::size_t level = max_level; level > 0; --level)
    {
        auto keep_subset = intersection(mesh[mesh_id_t::cells][level],
                                        mesh[mesh_id_t::all_cells][level - 1])
                           .on(level - 1);

        keep_subset.apply_op(coarsen_compatibility(m_tag));

        xt::xtensor_fixed<int, xt::xshape<dim>> stencil;
        for (std::size_t d = 0; d < dim; ++d)
        {
            stencil.fill(0);
            int grad_width = static_cast<int>(mesh_t::config::graduation_width);
            for (int s = -grad_width; s <= grad_width; ++s)
            {
```

```
59                 if (s != 0)
60                 {
61                     stencil[d] = s;
62                     auto subset = intersection(mesh[mesh_id_t::cells][level],
63                                                translate(mesh[mesh_id_t::cells][level - 1],stencil))
64                                       .on(level - 1);
65                     subset.apply_op(balance_2to1(m_tag, stencil));
66                 }
67             }
68         }
69     }
70
71     // Refinement grading procedure, implemented in Section 6.2.4
72     for (std::size_t level = max_level; level > min_level; --level)
73     {
74         auto subset_1 = intersection(mesh[mesh_id_t::cells][level],
75                                      mesh[mesh_id_t::cells][level]);
76         subset_1.apply_op(extend(m_tag));
77
78         static_nested_loop<dim, -1, 2>(
79             [&](auto stencil) {
80             auto subset = intersection(translate(mesh[mesh_id_t::cells][level], stencil),
81                                      mesh[mesh_id_t::cells][level-1]).on(level);
82
83             subset.apply_op(refinement_compatibility(m_tag));
84         });
85     }
86
87     for (std::size_t level = max_level; level > 0; --level)
88     {
89         auto keep_subset = intersection(mesh[mesh_id_t::cells][level],
90                                         mesh[mesh_id_t::all_cells][level - 1])
91                                .on(level - 1);
92         keep_subset.apply_op(coarsen\_compatibility(m_tag));
93     }
94
95     if (update_field_mr(m_field, field_old, m_tag)) // Implemented in Section 6.2.5
96     {
97         return true;
98     }
99     return false;
100 }
```

Let us comment on the most important parts of the previous long code.

- On Line 5, the `for_each_cell` executes what follows on every cell belonging to the category `cells` that has been specified. In particular, we see on Line 7 that every cell is—at the beginning of the process and until further modification—kept in the structure.
- Since we store the solution on the complete leaves, the first operation to perform is the computation of the details on them, in order to start the coarsening/refinement process. This is done on Line 15, where `compute_detail` (see Section 6.2.1 for its implementation) is applied to the leaves projected on the coarser level, in order to access siblings with their identifier $(\ell+1, 2\boldsymbol{k}+\boldsymbol{\delta})$ with $\boldsymbol{\delta} \in \Sigma$, where the role of ℓ is taken by `level`. Observe on Line 10 that since, at each loop in the refinement process, we exclude the finest level considered so far (*cf.* `ite`), we consider less and less levels to span as the computation goes on. Otherwise said, the emerged levels are progressively no longer taken into account.
- Then, we proceed to coarsen cells according to (2.31). On Lines 20 and 21 the level-wise threshold ϵ_ℓ given by (2.32) is computed. The operator `to_coarsen_mr` (*cf.* Line 27, see Section 6.2.2 for its implementation) is then applied to the leaves projected on the coarser level and uses the details computed right before.
- The coarsening is followed by the refinement by $\mathcal{H}_\epsilon$. The refinement based on the details—*cf.* (2.36)—is performed on Line 29. Using `regularity`, which is ν in the regularity $W^{\nu,\infty}$ of the solution, the refinement process is based on $\overline{\mu} = \min(\nu, 2\gamma+1)$ in (2.36) ($\gamma = 1$ in the code). The other refinement criterion (2.35) is applied on Line 38 and its implementation is detailed in Section 6.2.3.
- The last step before the actual modification of the mesh is the grading represented by $\mathcal{G}$. We start by enforcing a certain coherence between tags and mesh structure for the cells where the tag is `coarsen` on Line 50, with the operator `coarsen_compatibility`.
- Now that all the operations to obtain a new `tag` field representing the updated mesh are performed, the actual modification of the mesh and the numerical solution defined on it are performed on Line 95. Here, the procedure returns `true` if no modification has been done and the process can thus be stopped.

6.2.1 Details computation

The implementation of `compute_detail` to obtain the details for the case $d = 1$ and $d = 2$ is presented in the following code.

```
// 1D
template<class T, std::size_t order= T::mesh_t::config::prediction_order>
inline void operator()(Dim<1>, T &detail, const T &field) const
{
    auto qs_i = xt::eval(Qs_i<order>(field, level, i));
    detail(level+1, 2*i) = field(level+1, 2*i) - (field(level, i) + qs_i);
    detail(level+1, 2*i+1) = field(level+1, 2*i+1) - (field(level, i) - qs_i);
}

// 2D
template<class T, std::size_t order= T::mesh_t::config::prediction_order>
inline void operator()(Dim<2>, T &detail, const T &field) const
{
    auto qs_i = Qs_i<order>(field, level, i, j);
    auto qs_j = Qs_j<order>(field, level, i, j);
    auto qs_ij = Qs_ij<order>(field, level, i, j);

    detail(level+1, 2*i, 2*j) = field(level+1, 2*i, 2*j) - (field(level, i, j)+qs_i+qs_j-qs_ij);
    detail(level+1, 2*i+1, 2*j) = field(level+1, 2*i+1, 2*j) - (field(level, i, j) - qs_i+qs_j+qs_ij);
    detail(level+1, 2*i, 2*j+1) = field(level+1, 2*i, 2*j+1) - (field(level, i, j)+qs_i-qs_j+qs_ij);
    detail(level+1, 2*i+1, 2*j+1) = field(level+1, 2*i+1, 2*j+1) - (field(level, i, j)-qs_i-qs_j-qs_ij);
}
```

We observe that when calling `compute_detail` on a set of cells, the level of the cells as well as the indices are automatically available and used in the previous code. Moreover, the fact that `field` is a vector with several (q) components is automatically handled. The quantity `Qs_i<order>` on Line 5 just gives $Q_1^\gamma(k;\overline{\mathbf{f}}_\ell)$ from (2.16), with `order` playing the role of the number of neighbors γ in the prediction operator.

6.2.2 Mesh coarsening

Once the details are available, we can implement the function `to_coarsen_mr` that does what $\mathcal{T}_\epsilon$ indicates. We present the implementation in the case $d = 2$.

```
template<class T1, class T2>
inline void operator()(Dim<2>, const T1& detail, T2 &tag, double eps, std::size_t min_lev) const
{
    std::size_t fine_level = level + 1;
    if (fine_level > min_lev)
    {
        if (size == 1) // Scalar field
        {
            auto mask = (xt::abs(detail(fine_level, 2*i  ,   2*j)) < eps) and
                        (xt::abs(detail(fine_level, 2*i+1,   2*j)) < eps) and
                        (xt::abs(detail(fine_level, 2*i  , 2*j+1)) < eps) and
                        (xt::abs(detail(fine_level, 2*i+1, 2*j+1)) < eps);

            xt::masked_view(tag(fine_level, 2*i  ,   2*j), mask) = static_cast<int>(CellFlag::coarsen);
            xt::masked_view(tag(fine_level, 2*i+1,   2*j), mask) = static_cast<int>(CellFlag::coarsen);
            xt::masked_view(tag(fine_level, 2*i  , 2*j+1), mask) = static_cast<int>(CellFlag::coarsen);
            xt::masked_view(tag(fine_level, 2*i+1, 2*j+1), mask) = static_cast<int>(CellFlag::coarsen);
        }
        else // Vectorial field
        {
            auto mask = xt::sum((xt::abs(detail(fine_level, 2*i  ,   2*j)) < eps) and
                                (xt::abs(detail(fine_level, 2*i+1,   2*j)) < eps) and
                                (xt::abs(detail(fine_level, 2*i  , 2*j+1)) < eps) and
                                (xt::abs(detail(fine_level, 2*i+1, 2*j+1)) < eps), {1}) > (size-1);

            xt::masked_view(tag(fine_level, 2*i  ,   2*j), mask) = static_cast<int>(CellFlag::coarsen);
            xt::masked_view(tag(fine_level, 2*i+1,   2*j), mask) = static_cast<int>(CellFlag::coarsen);
            xt::masked_view(tag(fine_level, 2*i  , 2*j+1), mask) = static_cast<int>(CellFlag::coarsen);
            xt::masked_view(tag(fine_level, 2*i+1, 2*j+1), mask) = static_cast<int>(CellFlag::coarsen);
        }
    }
}
```

In this code, we first create a mask which detects the cells on which the coarsening criterion (2.31), *i.e.* where the detail of all (four) siblings is below the level-wise threshold ϵ_ℓ. Then, over the cells indexed by this mask, we put

the tag field, which shall be used to perform mesh adaptation, to `coarsen`.

6.2.3 Mesh refinement

Once the mesh coarsening is done, the refinement procedure $\mathcal{H}_\epsilon$ needs to be implemented as well. The first phase is the refinement based on the details following (2.36), which is the aim of `to_refine_mr`, which is implemented in the following code for $d = 1$.

```
1  template<class T1, class T2>
2  inline void operator()(Dim<1>, const T1& detail, T2 &tag, double eps, std::size_t max_level) const
3  {
4      constexpr auto size = T1::size;
5      std::size_t fine_level = level + 1;
6
7      if (fine_level < max_level)
8      {
9          if (size == 1) // Scalar field
10         {
11             auto mask = ((xt::abs(detail(fine_level, 2*i))) > eps) or
12                         ((xt::abs(detail(fine_level, 2*i+1))) > eps);
13             xt::masked_view(tag(fine_level, 2*i  ), mask) = static_cast<int>(CellFlag::refine);
14             xt::masked_view(tag(fine_level, 2*i+1), mask) = static_cast<int>(CellFlag::refine);
15         }
16         else // Vectorial field
17         {
18             auto mask = xt::sum(((xt::abs(detail(fine_level, 2*i  ))) > eps) or
19                                 ((xt::abs(detail(fine_level, 2*i+1))) > eps), {1}) > 0;
20             xt::masked_view(tag(fine_level, 2*i  ), mask) = static_cast<int>(CellFlag::refine);
21             xt::masked_view(tag(fine_level, 2*i+1), mask) = static_cast<int>(CellFlag::refine);
22         }
23     }
24 }
```

On Line 12, we detect siblings which detail exceeds $2^{\overline{\mu}+d}\epsilon_\ell$ (observe that when this function is called, `eps` here equals this value and not ϵ_ℓ). Then, the corresponding cells are tagged as to `refine`. To summarize, at this stage, the `tag` has been modified as follows:

$$\texttt{tag}_{\ell,\boldsymbol{k}} = \begin{cases} \texttt{refine}, & \text{if} \quad \max_{j\in[\![1,q]\!]} |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)| \in [2^{\overline{\mu}+d}\epsilon_\ell, +\infty[, \\ \texttt{keep}, & \text{if} \quad \max_{j\in[\![1,q]\!]} |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)| \in [\epsilon_\ell, 2^{\overline{\mu}+d}\epsilon_\ell[, \\ \texttt{coarsen}, & \text{if} \quad \max_{j\in[\![1,q]\!]} |\overline{\mathrm{d}}^j_{\ell,\boldsymbol{k}}(t)| \in [0, \epsilon_\ell[. \end{cases}$$

The second phase is the refinement based on (2.35), using the function `enlarge`. Its implementation for $d = 1$ is simply given by:

```
1  template<class T>
2  inline void operator()(Dim<1>, T &cell_flag) const
3  {
4      auto keep_mask = cell_flag(level, i) & static_cast<int>(CellFlag::keep);
5      for (int ii = -1; ii < 2; ++ii)
6          xt::masked_view(cell_flag(level, i + ii), keep_mask) |= static_cast<int>(CellFlag::enlarge);
7  }
```

Here, the neighboring cells to those which are to `keep` are set to `enlarge` and shall eventually be kept as well in the data structure.

6.2.4 Grading

The last step before actually modify the mesh is the grading $\mathcal{G}$ procedure. The first building block of this is the `coarsen_compatibility` operation, which reads, for $d = 1$:

```
1  template<class T>
2  inline void operator()(Dim<1>, T &field) const
3  {
4      xt::xtensor<bool, 1> mask =
5          (field(level + 1, 2 * i)     & static_cast<int>(CellFlag::keep)) |
```

```
        (field(level + 1, 2 * i + 1) & static_cast<int>(CellFlag::keep));
    xt::masked_view(field(level + 1, 2 * i), mask) |= static_cast<int>(CellFlag::keep);
    xt::masked_view(field(level + 1, 2 * i + 1), mask) |= static_cast<int>(CellFlag::keep);

    xt::masked_view(field(level, i), mask) |= static_cast<int>(CellFlag::keep);

    mask = (field(level + 1, 2 * i)     & static_cast<int>(CellFlag::coarsen)) &
           (field(level + 1, 2 * i + 1) & static_cast<int>(CellFlag::coarsen));
    xt::masked_view(field(level + 1, 2 * i), !mask) &= ~static_cast<int>(CellFlag::coarsen);
    xt::masked_view(field(level + 1, 2 * i + 1), !mask) &= ~static_cast<int>(CellFlag::coarsen);
    xt::masked_view(field(level, i), mask) |= static_cast<int>(CellFlag::keep);
}
```

If one of two siblings are to keep, then the mask will be true for both. Thus, both shall be kept. Moreover, their parent is also to keep. Then, if both siblings are to coarsen (one over two is not enough), the cells are coarsened and the parent cell is kept. If the opposite happens (one of the two siblings is not to coarsen), both are kept. This is followed by `balance_2to1_op` (here presented for $d = 2$):

```
template<class T, class stencil_t>
inline void operator()(Dim<2>, T &cell_flag, const stencil_t &stencil) const
{
    cell_flag(level, i - stencil[0], j - stencil[1])
    |= (cell_flag(level, i, j) &static_cast<int>(samurai::CellFlag::keep));
}
```

The idea here is that if a neighboring cell of the current cell is set to `keep`, it is the parent of some cells which have flagged it to `keep`. This implies that—in order to yield a graded mesh—the current cell needs to be kept as well. Then, we implemented the `extend` function, which for $d = 1$ reads:

```
template<class T>
inline void operator()(Dim<1>, T &tag) const
{
    auto refine_mask = tag(level, i) & static_cast<int>(samurai::CellFlag::refine);
    for (int ii = -1; ii < 2; ++ii)
    {
        xt::masked_view(tag(level, i + ii), refine_mask) |= static_cast<int> (samurai::CellFlag::keep);
    }
}
```

The aim of this is to enforce that if we refine a cell, we cannot coarsen its neighboring cell, because otherwise we would loose the grading property. Finally, one has `refinement_compatibility`:

```
template<class T>
inline void operator()(Dim<1>, T &tag) const
{
    auto i_even = i.even_elements();
    if (i_even.is_valid())
    {
        auto mask = tag(level, i_even) & static_cast<int>(CellFlag::keep);
        xt::masked_view(tag(level-1, i_even>>1), mask) |= static_cast<int>(CellFlag::refine);
    }
    auto i_odd = i.odd_elements();
    if (i_odd.is_valid())
    {
        auto mask = tag(level, i_odd) & static_cast<int>(CellFlag::keep);
        xt::masked_view(tag(level-1, i_odd>>1), mask) |= static_cast<int>(CellFlag::refine);
    }
}
```

This rationale behind this is the following. If a cell is set to `refine`, this comes from the fact that its neighbor is to `keep`. This means that the parent of this neighbor needs to be refined.

6.2.5 Mesh and field update

Once the `tag` field is ready, it is used to create the new mesh as well as to adapt the numerical solution on the new mesh. This is what is implemented in `update_field_mr`, which C++ code reads as follows.

```
auto & mesh = field.mesh();
```

```
2
3  for_each_interval(mesh[mesh_id_t::cells], [&](std::size_t level, const auto& interval,const auto& index)
4  {
5      for (auto i=interval.start; i<interval.end; ++i)
6      {
7          if ( tag[i + interval.index] & static_cast<int>(CellFlag::refine))
8          {
9              static_nested_loop<dim-1, 0, 2>([&](auto& stencil)
10             {
11                 auto new_index = 2*index + stencil;
12                 cl[level + 1][new_index].add_interval({2*i, 2*i + 2});
13             });
14         }
15         else if ( tag[i + interval.index] & static_cast<int>(CellFlag::keep))
16         {
17             cl[level][index].add_point(i);
18         }
19         else
20         {
21             cl[level - 1][index >> 1].add_point(i >> 1);
22         }
23     }
24 });
25
26 mesh_t new_mesh = {cl, mesh.min_level(), mesh.max_level()};
27 if (mesh == new_mesh)
28     return true;
29
30 Field new_field("new_f", new_mesh);
31 new_field.fill(0);
32
33 for(std::size_t level = min_level; level <= max_level; ++level)
34 {
35     auto set = intersection(mesh[mesh_id_t::cells][level],
36                             new_mesh[mesh_id_t::cells][level]);
37     set.apply_op(copy(new_field, field));
38 }
39
40 for(std::size_t level = min_level + 1; level <= max_level; ++level)
41 {
42     auto set_coarsen = samurai::intersection(mesh[mesh_id_t::cells][level],
43                                              new_mesh[mesh_id_t::cells][level-1])
44                            .on(level - 1);
45     set_coarsen.apply_op(projection(new_field, field));
46     auto set_refine = intersection(new_mesh[mesh_id_t::cells][level],
47                                    mesh[mesh_id_t::cells][level-1])
48                           .on(level - 1);
49     set_refine.apply_op(prediction<pred_order, true>(new_field, field));
50 }
51
52 auto old_mesh = old_field.mesh();
53 for (std::size_t level = min_level; level <= max_level; ++level)
54 {
55     auto subset = intersection(intersection(old_mesh[mesh_id_t::cells][level],
56                                     difference(new_mesh[mesh_id_t::cells][level], mes[mesh_id_t::cells][level])),
57                                     mesh[mesh_id_t::cells][level-1]).on(level);
58
59     subset.apply_op(copy(new_field,  old_field));
60 }
61
62 field.mesh_ptr()->swap(new_mesh);
63 old_field.mesh_ptr()->swap(new_mesh);
64 std::swap(field.array(), new_field.array());
65 std::swap(old_field.array(), new_field.array());
66
67 return false;
```

Line 1 recovers the current mesh directly from the solution field, which contains a pointer to the mesh it is associated with.

- The first step is to create the new mesh according to the `tag` field. The `for_each_interval` command on Line 3 allows to cycle over all the intervals of category `cells` that has been specified. The interval is broken into its subcells on Line 5 and—if a cell is tagged to `refine` (*cf.* Line 7)—its children are added to the `CellList` of the new mesh to create, *cf.* Line 12. In this part, we take advantage of a `static_nested_loop` on Line 9 in order to write a code working irrespective of the choice of d. When a cell is flagged as `keep`, see Line 15, it is added to the `CellList`. Otherwise, the cell at a coarser level is added. The new mesh is created on Line 26 and if nothing is done, the whole procedure returns `true` to indicate that everything in the process can stop here. In particular, a new mesh where the cells of category `cells` are those in the `CellList` named `cl` is created.

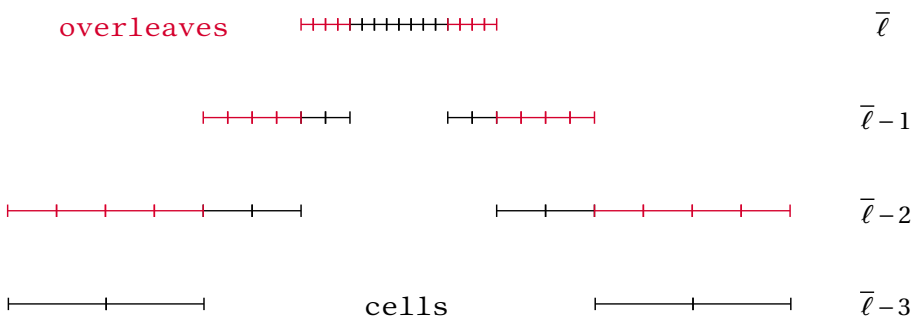


Figure 6.2: Illustration of what the `overleaves` (in red) are, once the leaves `cells` (in black) are given.

- A second step is to adapt the solution to the new mesh. The new field is created on Line 30. For the cells which were present both in the old and new mesh, *cf.* Line 36, data are just copied, see Line 37. For the cells which have been coarsened, *cf.* Line 44, the projection operator $\boldsymbol{P}_{\triangledown}$ is applied on Line 45 to yield the average of the siblings on the old mesh over the parent on the new mesh. Finally, for the cells that have been refined, *cf.* Line 48, the prediction operator $\boldsymbol{P}_{\triangle}$ is used to construct lacking information, see Line 49.

The process terminates by swapping the pointers to the mesh structure and to the linear structure for the storage of the numerical solution on the mesh.

6.3 Level jumps, reconstruction flattening and lattice Boltzmann schemes

We explain how we solve—with a suboptimal choice—the issue exposed in Section 2.5.4.1 concerning an efficient non-recursive implementation of the reconstruction operator. Troubles come when the cells needed by the reconstruction flattening at some level are beneath cells of finer level, thus the value of the solution they carry is not accurate enough. The solution that we have envisioned—thanks to the grading of the tree associated with the mesh—consists in performing the stream computation a finer level $\ell+1$ instead of ℓ (except when $\ell=\overline{\ell}$), using the reconstruction flattening carelessly because we know that information is accurate enough, and then projecting on the leaves. The result is the same, except for the fact that the computations use reliable information within the tolerance ϵ, because the projection simplifies inter-cell fluxes between couples of cells. This process is where the `overleaves`—which are illustrated using Figure 6.2—come into play.

6.3.1 Overleaves

Recall that the `overleaves` do not take part in the process of mesh adaptation and the values stored on them could be not up-to-date. Therefore, before applying the numerical scheme, the values stored on the `overleaves` are updated with the following procedure.

```
1 for (std::size_t level = min_level + 1; level <= max_level; ++level)
2 {
3     auto overleaves_to_predict = difference(difference(mesh[mesh_id_t::overleaves][level],
4                                                        mesh[mesh_id_t::cells_and_ghosts[level]),
5                                             mesh[mesh_id_t::proj_cells][level]);
6
7     overleaves_to_predict.apply_op(prediction<1, false>(field));
8 }
```

In this code, the only overleaves on which the solution is allegedly not up-to-date (*cf.* Line 5) are those which are neither `cells_and_ghosts` nor `proj_cells` cells, which have been previously updated. On these `overleaves`, the solution is updated using the prediction operator (in this code, with $\gamma=1$), see Line 7.

6.3.2 Reconstruction flattening

We provide the example for a D_1Q_2 scheme where we need to compute the flattening coefficients $(F^j_{\Delta\ell,\boldsymbol{\delta}})_{\boldsymbol{\delta}\in\Xi^j_{\Delta\ell}}$ and shifts $\Xi^j_{\Delta\ell}$—see Section 2.5.4.1 and (2.45)—for $q = 2$ discrete velocities

```
1  template<class coord_index_t>
2  auto compute_reconstruction(std::size_t min_level, std::size_t max_level)
3  {
4      coord_index_t i = 0;
5      std::vector<std::vector<reconstruction_map<coord_index_t>>> data(max_level-min_level+1);
6
7      for(std::size_t dl=0; dl<max_level-min_level+1; ++dl)
8      {
9          int size = (1<<dl);
10         data[dl].resize(2);
11
12         data[dl][0] = reconstruction(dl, i*size - 1) - reconstruction(dl, (i+1)*size - 1);
13         data[dl][1] = reconstruction(dl, (i+1)*size) - reconstruction(dl, i*size);
14     }
15     return data;
16 }
```

The computation of the weights and shifts for the positive velocity is on Line 12, whereas the ones for the negative velocity is done on Line 13. We construct a vector with double indices where the first one concerns $\Delta\ell$ and the second one the discrete velocity to consider. The fact of writing Line 4 means that we center the computation of the flattening coefficients around the current cell, considering the pseudo-fluxes at its interfaces. `reconstruction` is implemented recursively—for $\gamma = 1$—as:

```
1  template<class index_t>
2  auto reconstruction(std::size_t dl, const index_t &i, bool reset=false)
3  {
4      static std::map<std::tuple<std::size_t, index_t>, reconstruction_map<index_t>> values;
5      if (reset)
6          values.clear();
7      if (dl == 0)
8          return reconstruction_map<index_t>{i};
9
10     auto iter = values.find({dl, i});
11     if (iter == values.end())
12     {
13         auto ig = i >> 1;
14         double d_x = (i & 1)? -1./8: 1./8;
15
16         return values[{dl, i}] = reconstruction(dl-1, ig) - d_x*(reconstruction(dl-1, ig+1)
17                                                                  -reconstruction(dl-1, ig-1));
18     }
19     else
20         return iter->second;
21 }
```

The map that we create link couples of level jump and shifts looking for neighboring cells to the weights. Observe that at the finest level, thus for $\Delta\ell = 0$, see Line 8, we put the coefficient to one by default. Otherwise, see Line 11, if the weights and the shifts do not have been computed, we apply the prediction operator $\boldsymbol{P}_\triangle$. Before the simulation and once for all, the reconstruction coefficients and shifts are computed and stored using:

```
1  auto rec_coeff = compute_reconstruction<coord_index_t>(min_level, max_level);
```

for the given minimum and maximum level that the user has specified for the numerical simulation at hand.

6.3.3 Lattice Boltzmann schemes

In order to illustrate how the adaptive lattice Boltzmann is finally implemented using the `overleaves`, we use the D_1Q_2 as example. On the one hand, the collision phase is done on the complete leaves of the adaptive tree, that is on the `cells`.

```
1  for (std::size_t level = min_level; level <= max_level; ++level)    {
```

```
    auto leaves = samurai::intersection(mesh[mesh_id_t::cells][level],
                                        mesh[mesh_id_t::cells][level]);
    leaves([&](auto &interval, auto) {
        auto k = interval;
        auto m1   = xt::eval(        f(0, level, k) + f(1, level, k));
        auto m2   = xt::eval(lambda*(f(0, level, k) - f(1, level, k)));
        auto m2 = (1 - s2) * m2 + s2*e2*m1;
        f_star(0, level, k) = 0.5*m1+0.5/lamba*m2;
        f_star(1, level, k) = 0.5*m1-0.5/lamba*m2;
    });
}
```

For every level between $\underline{\ell}$ and $\overline{\ell}$, *cf.* Line 1, we perform the collision on the complete leaves of the adaptive tree, *cf.* Line 3. This is done by first changing the basis through $\boldsymbol{M}$, *cf.* Line 7, colliding for the non-conserved moment, *cf.* Line 8 and eventually apply $\boldsymbol{M}^{-1}$, see Line 10. On the other hand, the stream phase is encoded as follows.

```
1  auto h_ovrlvs = samurai::make_field<double, 2>("help_f", mesh);
2  h_ovrlvs.fill(0.);
3
4  for (std::size_t level = 0; level <= max_level; ++level)
5  {
6      if (level == max_level) {
7          for_each_interval(mesh[mesh_id_t::cells], [&](std::size_t level, const auto& interval,const auto&)
8          {
9              auto k = interval;
10             f(0, max_level, k) = xt::eval(f_star(0, max_level, k - 1));
11             f(1, max_level, k) = xt::eval(f_star(1, max_level, k + 1));
12         });
13     }
14     else    {
15         std::size_t dlp1 = max_level - (level + 1);
16         double coeff = 1. / (1 << dlp1);
17
18         auto overleaves = samurai::intersection(mesh[mesh_id_t::cells][level],
19                                                 mesh[mesh_id_t::cells][level]).on(level + 1);
20
21         overleaves([&](auto &interval, auto) {
22             auto k = interval;
23             auto fp = xt::eval(f_star(0, level + 1, k));
24             auto fm = xt::eval(f_star(1, level + 1, k));
25             for(auto &c: rec_coeff[j][0].coeff)
26             {
27                 coord_index_t stencil = c.first;
28                 double weight = c.second;
29                 fp += coeff * weight * f_star(0, level + 1, k + stencil);
30             }
31             for(auto &c: pred_coeff[j][1].coeff)
32             {
33                 coord_index_t stencil = c.first;
34                 double weight = c.second;
35                 fm += coeff * weight * f_star(1, level + 1, k + stencil);
36             }
37             h_ovrlvs(0, level + 1, k) = fp;
38             h_ovrlvs(1, level + 1, k) = fm;
39         });
40
41         for_each_interval(mesh[mesh_id_t::cells], [&](std::size_t level, const auto& interval,const auto&)
42         {
43             auto k = interval;
44             f(0, level, k) = xt::eval(0.5*(h_ovrlvs(0, level + 1, 2*k) + h_ovrlvs(0, level + 1, 2*k + 1)));
45             f(1, level, k) = xt::eval(0.5*(h_ovrlvs(1, level + 1, 2*k) + h_ovrlvs(1, level + 1, 2*k + 1)));
46         });
47     }
48 }
```

On Line 1, we create a field that we shall use to average the solution back from the overleaves to the leaves. If the current level of resolution the finest one $\overline{\ell}$, see Line 6, the reference scheme is applied, *cf.* Line 11. If the cells are not at the finest level, see Line 14, the `overleaves` need to be used. They are nothing else than `cells` projected at a finer level, see Line 19. Here, we use the flattened coefficients that have been previously computed, see for example Line 29, as we were performing the numerical scheme at a level $\ell + 1$, updating the solution on the `overleaves`. Eventually, since the solution has to be updated on the complete leaves `cells`, the solution obtained on the overaleaves is averaged back on the `cells` underneath, see Line 44.

6.4 Conclusions

In Chapter 6, we have presented how multiresolution is implemented using SAMURAI and how we cope with the efficient implementation of the reconstruction operator, which is used in the adaptive lattice Boltzmann schemes presented before. The material of Chapter 6 is only a specific and partial demonstration of the possibilities offered by SAMURAI: for more information, the interested reader can consult the documentation https://hpc-math-samurai.readthedocs.io. Concerning the perspectives of this work, the code parallelization is surely one of the most interesting topics to be studied in forthcoming contributions. Also, as GPUs have proved to be extremely efficient supports for lattice Boltzmann computations, a future axis of research concerns the implementation of these schemes, within SAMURAI, on this type of architecture. It should be noted that a large body of literature exists on parallel [Axner et al., 2008, Mazzeo and Coveney, 2008, Vidal et al., 2010] and GPU implementations [Bernaschi et al., 2010, Tomczak and Szafran, 2019] of the lattice Boltzmann method on complex geometries. Therefore, all future effort in this direction has to take these previous works into account. We also stress that we adopted a suboptimal strategy using the `overleaves` which—though only needed for the computation of the pseudo-fluxes for the adaptive scheme—are permanently present in the data structure. A next step in the code optimization would be to find a more efficient way of evolving the solution using reliable data without having to use these `overleaves`.

Part III

Numerical analysis of lattice Boltzmann schemes

General introduction

As seen in Chapter 1, lattice Boltzmann methods act in a kinetic fashion by employing a certain number of discrete velocities, larger than the number of macroscopic equations to be solved. The scheme proceeds *via* two distinct steps. The first one is a local non-linear collision phase on each site of the mesh, followed by a lattice-constrained transport, which is inherently linear. The local nature of the collision phase makes the method embarrassingly parallel and the fact that the "particles" are constrained to dwell on the lattice allows to implement the stream phase as a pointer shift in memory. This results in a very efficient numerical method capable of reaching problems of important size in terms of computational and memory cost. To our understanding, the highest price to pay for this highly efficient implementation of the method is the lack of a unified convergence theory—which would allow understanding why the overall procedure works well at approximating the solution of the target problem. Rigorous proofs of convergence have been possible only for either very simple schemes or featuring some very specific structure. This is essentially due to the fact that—the standpoint of the lattice Boltzmann schemes being kinetic—the number of discrete velocities q is larger than the number of equations to solve N. The lack of a unified convergence theory comes from the fact that we do not have precise notions of consistency and stability for lattice Boltzmann schemes. In the spirit of the Lax theorem for Finite Difference schemes, consistency, and stability allow to show convergence but are generally easier to check than this latter property.

Aim and structure of Part III

The aim of Part III is to introduce rigorous notions of consistency and stability for lattice Boltzmann schemes, which allow us to analyze their convergence towards the solution of a given problem. Otherwise said, Part III has the ambitious target of shifting lattice Boltzmann back to the realm of numerical analysis. It is structured as follows. In Chapter 7, we eliminate the non-conserved moments from the lattice Boltzmann schemes at the fully discrete level, in order to recast the method into the form of multi-step Finite Difference schemes on the conserved moments only. This allows to keep only the N variables of interest for the conservation laws to solve and allows to forget about the lattice Boltzmann methods being discretizations of mesoscopic equations. The notions of consistency and stability are therefore naturally inherited from those of multi-step Finite Difference schemes. Once these notions are established, we aim at studying them on the original lattice Boltzmann scheme without having to explicitly perform the transformation towards the Finite Difference schemes. From this standpoint, consistency is studied in Chapter 8, whereas linear L^2 stability is analyzed in Chapter 9. Finally, Chapter 10 deals with the study of the initialisation of lattice Boltzmann schemes, taking into account that they feature more unknowns q than the initial data at our disposal, namely N.

Published works

The content of Part III has led to the following publications in peer-reviewed journals.

- [Bellotti et al., 2022e] Bellotti, T., Graille, B., and Massot, M. (2022e). Finite difference formulation of any lattice Boltzmann scheme. *Numerische Mathematik*, 152:1–40.
  This covers the content of Chapter 7 and part of Chapter 9.
- [Bellotti, 2023b] Bellotti, T. (2023b). Truncation errors and modified equations for the lattice Boltzmann method via the corresponding Finite Difference schemes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 57(3):1225–1255.
  This spans the content of Chapter 8.
- [Bellotti, 2023a] Bellotti, T. (2023a). Initialisation from lattice Boltzmann to multi-step Finite Difference methods: modified equations and discrete observability. *Submitted, see https://hal.science/hal-03989355*.
  This preprint covers the content of Chapter 10.

Chapter 7

Elimination of the non-conserved moments: corresponding Finite Difference schemes

General context and motivation

As previously pointed out—from our standpoint—the elimination of the non-conserved moments in the lattice Boltzmann schemes is the keystone to be able to perform rigorous numerical analysis on this class of numerical methods. Non-conserved moments, while playing a key role in the construction of the numerical algorithm, as in relaxation schemes, do not have a counterpart in the system of equations to be solved at the continuous macroscopic level.

State of the art

In the past, some authors have noticed that for some particular lattice Boltzmann schemes, one has a corresponding (sometimes called "equivalent") Finite Difference formulation on the conserved variables. Despite this, no general theory has been formulated. For instance: in [Suga, 2010], the author derives—by direct computations—a three-stages Finite Difference scheme from D_1Q_3 scheme with one conserved moment, limiting the discussion to a linear framework with one relaxation parameter. Similarly, Dellacherie [Dellacherie, 2014] derives a three-stages (two-steps) Finite Difference scheme for the D_1Q_2 lattice Boltzmann scheme with one conserved moment. Again, this is limited to one spatial dimension and to a linear framework. A higher level of generality has been reached by the works of Ginzburg and collaborators, see [Ginzburg, 2009] for a recap. They succeeded, using a link formalism, in writing a class of Lattice Boltzmann schemes as Finite Difference schemes [d'Humières and Ginzburg, 2009]. With their highly constrained link structure to be enforced, the resulting Finite Difference scheme with three stages is valid regardless of the spatial dimension and the choice of discrete velocities. The limitation is that the structure of the scheme is heavily constrained: the evolution equation of the moving particles can depend on the distribution of the rest particles only *via* the conserved moments the equilibria depend upon and the schemes must be two-relaxation-times models with collision operator shaped by the link structure and "magic parameter" equal to one-fourth for every link. Even if [Ginzburg, 2009] explicitly provides the expression of the Finite Difference scheme only for one conserved moment, the proposed procedure works for any number of conserved moments. This is made possible by showing that each distribution function at the new time $t + \Delta t$ can be written as a function of itself at time $t - \Delta t$ (two steps before) plus known terms depending on the equilibria. The difficulty in establishing a result for more general schemes comes from the coupling between spatial operators and time shifts. Finally, the authors of [Lin et al., 2021] have performed essentially the same computation as [Suga, 2010] on a D_1Q_3 with one conserved moment and multiple relaxation parameters. We must mention that during the thesis, an interesting work by [Fučík and Straka, 2021] has been published covering the very same subject and essentially coming to the main result of Chapter 7. Their focus is different than ours since they adopt a purely algorithmic approach rather than a precise algebraic characterization of lattice Boltzmann schemes. We actually provide more insight into the bound on the number of time steps of the corresponding Finite Difference scheme and our formalism, based on

polynomials, aims at providing a direct link with the classical tools for consistency—*cf.* Chapter 8—and for the stability analysis—*cf.* Chapter 9. In [Fučík and Straka, 2021], the authors rely on a decomposition of the scheme using an hollow matrix (matrix with zero entries on the diagonal) yielding an equivalent form of the scheme with the diagonal non-equilibrium part, after a finite number of steps of their algorithm. However, to the best of our understanding, the origin of such algorithm is not fully clear. In their work, the spatial shifts of data introduced by the stream phase are taken into account using a rather cumbersome system of indices, whereas we rely on an simpler algebraic characterization of the stream phase.

Aims and structure of Chapter 7

The aim of Chapter 7 is to do—for general schemes and relying on a classical algebraic framework—what the previously cited works did for specific ones: eliminate the non-conserved moments from lattice Boltzmann schemes in order to obtain multi-step Finite Difference schemes on the conserved moments only. To this end, Chapter 7 is structured as follows. Section 7.1 proposes some introductory examples to understand the process of elimination of the non-conserved moments on specific schemes. This allows to understand the basic way of proceeding. Then, Section 7.2 discusses the analogous of the elimination of the non-conserved moments on linear systems of Ordinary Differential Equations (ODEs), in order to clarify which theorem and assumptions are needed to do the same on lattice Boltzmann schemes in full generality. The theorem to employ is indeed the celebrated Cayley-Hamilton theorem on commutative rings, which is introduced and discussed in Section 7.3. Therefore, the objective of Section 7.4 is to construct a specific commutative ring to represent the lattice Boltzmann schemes. In the pivotal Section 7.5, we use the Cayley-Hamilton theorem to eliminate the non-conserved moments from any lattice Boltzmann scheme, yielding what we call "corresponding Finite Difference schemes", which are multi-step. Section 7.6 is devoted to analyze the number of steps in the corresponding Finite Difference schemes more closely, taking the opportunity to present examples and possible simplifications of the problem. In Section 7.7, we observe that having recast the evolution of the discrete solution for the conserved moments as multi-step Finite Difference schemes, the concepts of consistency and stability are directly inherited from those for these latter schemes. This allows to construct a convergence theory like the one from the Lax theorem [Lax and Richtmyer, 1956]. Other results on Finite Difference schemes can be used as well. This is illustrated with the help of a particular lattice Boltzmann scheme. We eventually conclude and bridge with the forthcoming Chapters in Section 7.8.

Contents

7.1 First examples

We start with simple one-dimensional examples with few discrete velocities, generalizing the computations by [Dellacherie, 2014]. This is done in order to understand which kind of algebraic manipulations on the scheme are needed to eliminate the non-conserved moments.

7.1.1 D_1Q_2 scheme with one conserved moment

We consider the scheme introduced in Section 1.5.1. It can be recast on the moments in the form, where $t \in \Delta t\mathbb{N}$ and $x \in \Delta x\mathbb{Z}^d$:

$$\mathsf{m}_1(t+\Delta t,x) = \frac{1}{2}(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x)) + \frac{1-s_2}{2\lambda}(\mathsf{m}_2(t,x-\Delta x)-\mathsf{m}_2(t,x+\Delta x)) + \frac{s_2}{2\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))), \tag{7.1}$$

$$\mathsf{m}_2(t+\Delta t,x) = \frac{\lambda}{2}(\mathsf{m}_1(t,x-\Delta x)-\mathsf{m}_1(t,x+\Delta x)) + \frac{1-s_2}{2}(\mathsf{m}_2(t,x-\Delta x)+\mathsf{m}_2(t,x+\Delta x)) + \frac{s_2}{2}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))+m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))). \tag{7.2}$$

The first equation (7.1) concerns the conserved moment, whereas the second one (7.2) gives the evolution of the non-conserved moment. The aim is to eliminate the non-conserved moment m_2 from (7.1). To do so, consider (7.2) at the shifted time $t-\Delta t$ and at points $x \pm \Delta x$:

$$\mathsf{m}_2(t,x-\Delta x) = \frac{\lambda}{2}(\mathsf{m}_1(t-\Delta t,x-2\Delta x)-\mathsf{m}_1(t-\Delta t,x)) + \frac{1-s_2}{2}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)+\mathsf{m}_2(t-\Delta t,x)) + \frac{s_2}{2}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))+m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))).$$

$$\mathsf{m}_2(t,x+\Delta x) = \frac{\lambda}{2}(\mathsf{m}_1(t-\Delta t,x)-\mathsf{m}_1(t-\Delta t,x+2\Delta x)) + \frac{1-s_2}{2}(\mathsf{m}_2(t-\Delta t,x)+\mathsf{m}_2(t-\Delta t,x+2\Delta x)) + \frac{s_2}{2}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))+m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))).$$

Taking the difference of these two equations gives

$$\mathsf{m}_2(t,x-\Delta x)-\mathsf{m}_2(t,x+\Delta x) = \frac{\lambda}{2}(\mathsf{m}_1(t-\Delta t,x-2\Delta x)-2\mathsf{m}_1(t-\Delta t,x)+\mathsf{m}_1(t-\Delta t,x+2\Delta x)) + \frac{1-s_2}{2}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x)) + \frac{s_2}{2}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))). \tag{7.3}$$

In this equality, m_2 is still present. Consider (7.1) at the previous time step $t-\Delta t$ and at the points $x \pm \Delta x$ of the lattice. This gives

$$\mathsf{m}_1(t,x-\Delta x) = \frac{1}{2}(\mathsf{m}_1(t-\Delta t,x-2\Delta x)+\mathsf{m}_1(t-\Delta t,x)) + \frac{1-s_2}{2\lambda}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x)) + \frac{s_2}{2\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))).$$

$$\mathsf{m}_1(t,x+\Delta x) = \frac{1}{2}(\mathsf{m}_1(t-\Delta t,x)+\mathsf{m}_1(t-\Delta t,x+2\Delta x)) + \frac{1-s_2}{2\lambda}(\mathsf{m}_2(t-\Delta t,x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x)) + \frac{s_2}{2\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))).$$

Summing these two equations provides

$$\begin{aligned}\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x) = \frac{1}{2}(\mathsf{m}_1(t-\Delta t,x-2\Delta x)+2\mathsf{m}_1(t-\Delta t,x)+\mathsf{m}_1(t-\Delta t,x+2\Delta x))\\ +\frac{1-s_2}{2\lambda}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x))\\ +\frac{s_2}{2\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))).\end{aligned}$$

In this expression, we isolate the term that we wish to eliminate from (7.3). This reads

$$\begin{aligned}\frac{1-s_2}{2}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x)) = \lambda(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))\\ -\frac{\lambda}{2}(\mathsf{m}_1(t-\Delta t,x-2\Delta x)+2\mathsf{m}_1(t-\Delta t,x)+\mathsf{m}_1(t-\Delta t,x+2\Delta x))\\ -\frac{s_2}{2}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))),\end{aligned}$$

which put into (7.3) yields

$$\mathsf{m}_2(t,x-\Delta x)-\mathsf{m}_2(t,x+\Delta x) = \lambda(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))-2\lambda\mathsf{m}_1(t-\Delta t,x). \tag{7.4}$$

Inserting into (7.1) leads to

$$\begin{aligned}\mathsf{m}_1(t+\Delta t,x) = \frac{1}{2}(2-s_2)(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))-(1-s_2)\mathsf{m}_1(t-\Delta t,x)\\ +\frac{s_2}{2\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))),\end{aligned} \tag{7.5}$$

which is a Finite Difference scheme solely on the conserved moment m_1 linking three time stages $t+\Delta t$, t and $t-\Delta t$. The scheme by (7.5) is the same found in [Dellacherie, 2014] under the assumption that m_3^{eq} is a linear function of m_1. This scheme can be interpreted as follows:

- For $s_2 \in]0,1]$, this is a convex combination (indeed, a θ-scheme) of the scheme

  $$\mathsf{m}_1(t+\Delta t,x) = \mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x)-\mathsf{m}_1(t-\Delta t,x),$$

  for $s_2 = 0$, which is consistent with the wave equation with wave velocities $\pm\lambda$, and the Lax-Friedrichs scheme

  $$\mathsf{m}_1(t+\Delta t,x) = \frac{1}{2}(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))+\frac{1}{2\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))),$$

  for $s_2 = 1$.
- For $s_2 \in]1,2]$, this is a convex combination of a Lax-Friedrichs scheme ($s_2 = 1$) and a leap-frog scheme ($s_2 = 2$)

  $$\mathsf{m}_1(t+\Delta t,x) = \mathsf{m}_1(t-\Delta t,x)+\frac{1}{\lambda}(m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_2^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))).$$

7.1.2 $\mathrm{D}_1\mathrm{Q}_3$ scheme with two conserved moments

We consider the scheme introduced in Section 1.5.2 with moment matrix $\boldsymbol{M}$ given by (1.5) and two conserved moments, namely $N = 2$. For the sake of presentation, we assume that m_3^{eq} depends only on m_1. This assumption aims only at dealing with shorter formulæ. Letting $t \in \Delta t\mathbb{N}$ and $x \in \Delta x\mathbb{Z}^d$, the scheme on the moments reads

$$\begin{aligned}\mathsf{m}_1(t+\Delta t,x) = \mathsf{m}_1(t,x)+\frac{1}{2\lambda}(\mathsf{m}_2(t,x-\Delta x)-\mathsf{m}_2(t,x+\Delta x))\\ +\frac{1-s_3}{2\lambda^2}(\mathsf{m}_3(t,x-\Delta x)-2\mathsf{m}_3(t,x)+\mathsf{m}_3(t,x+\Delta x))\\ +\frac{s_3}{2\lambda^2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))),\end{aligned} \tag{7.6}$$

for the first conserved moment m_1, and

$$\mathsf{m}_2(t+\Delta t,x)=\frac{1}{2}(\mathsf{m}_2(t,x-\Delta x)+\mathsf{m}_2(t,x+\Delta x))+\frac{1-s_3}{2\lambda}(\mathsf{m}_3(t,x-\Delta x)-\mathsf{m}_3(t,x+\Delta x))\\+\frac{s_3}{2\lambda}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))),\tag{7.7}$$

for the second conserved moment m_2. Finally, for the non-conserved moment m_3, we have

$$\mathsf{m}_3(t+\Delta t,x)=\frac{\lambda}{2}(\mathsf{m}_2(t,x-\Delta x)-\mathsf{m}_2(t,x+\Delta x))+\frac{1-s_3}{2}(\mathsf{m}_3(t,x-\Delta x)+\mathsf{m}_3(t,x+\Delta x))\\+\frac{s_3}{2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))).\tag{7.8}$$

The aim is to eliminate m_3 both from (7.6) and (7.7). Still, we do not want to eliminate m_2 from (7.6) and m_1 from (7.7) because both these moments are conserved and should therefore remain in the final system. Looking at (7.7) compared to (7.1) and at (7.8) compared to (7.2), the Finite Difference scheme on m_2 can be found by exactly the same computation as Section 7.1.1 and reads

$$\mathsf{m}_2(t+\Delta t,x)=\frac{1}{2}(2-s_3)(\mathsf{m}_2(t,x-\Delta x)+\mathsf{m}_2(t,x+\Delta x))-(1-s_3)\mathsf{m}_2(t-\Delta t,x)\\+\frac{s_3}{2\lambda}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))).$$

The computation of the Finite Difference scheme for m_1 needs to be done differently. Consider (7.8) at the previous time $t-\Delta t$ and on the points $x-\Delta x$, x and $x+\Delta x$:

$$\mathsf{m}_3(t,x-\Delta x)=\frac{\lambda}{2}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x))+\frac{1-s_3}{2}(\mathsf{m}_3(t-\Delta t,x-2\Delta x)+\mathsf{m}_3(t-\Delta t,x))\\+\frac{s_3}{2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))).\tag{7.9}$$

$$\mathsf{m}_3(t,x)=\frac{\lambda}{2}(\mathsf{m}_2(t-\Delta t,x-\Delta x)-\mathsf{m}_2(t-\Delta t,x+\Delta x))+\frac{1-s_3}{2}(\mathsf{m}_3(t-\Delta t,x-\Delta x)+\mathsf{m}_3(t-\Delta t,x+\Delta x))\\+\frac{s_3}{2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+\Delta x))).\tag{7.10}$$

$$\mathsf{m}_3(t,x+\Delta x)=\frac{\lambda}{2}(\mathsf{m}_2(t-\Delta t,x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x))+\frac{1-s_3}{2}(\mathsf{m}_3(t-\Delta t,x)+\mathsf{m}_3(t-\Delta t,x+2\Delta x))\\+\frac{s_3}{2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))).\tag{7.11}$$

Considering (7.9) -2(7.10) $+$ (7.11) (making us think of the three-points scheme for the Laplace operator) provides

$$\begin{aligned}&\mathsf{m}_3(t,x-\Delta x)-2\mathsf{m}_3(t,x)+\mathsf{m}_3(t,x+\Delta x)\\&\quad=\frac{\lambda}{2}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-2\mathsf{m}_2(t-\Delta t,x-\Delta x)+2\mathsf{m}_2(t-\Delta t,x+\Delta x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x))\\&\qquad+\frac{1-s_3}{2}(\mathsf{m}_3(t-\Delta t,x-2\Delta x)-2\mathsf{m}_3(t-\Delta t,x-\Delta x)+2\mathsf{m}_3(t-\Delta t,x)\\&\qquad\quad-2\mathsf{m}_3(t-\Delta t,x+\Delta x)+\mathsf{m}_3(t-\Delta t,x+2\Delta x))\\&\qquad+\frac{s_3}{2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-\Delta x))+2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))\\&\qquad\quad-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))).\end{aligned}\tag{7.12}$$

We still have to handle the third and fourth lines out of the previous expression. To do this, consider (7.6) at time step $t-\Delta t$ at points $x\pm\Delta x$, giving

$$\begin{aligned}\mathsf{m}_1(t,x-\Delta x) = \mathsf{m}_1(t-\Delta t,x-\Delta x) &+ \frac{1}{2\lambda}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x))\\ &+ \frac{1-s_3}{2\lambda^2}(\mathsf{m}_3(t-\Delta t,x-2\Delta x)-2\mathsf{m}_3(t-\Delta t,x-\Delta x)+\mathsf{m}_3(t-\Delta t,x))\\ &+ \frac{s_3}{2\lambda^2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))),\end{aligned}$$

$$\begin{aligned}\mathsf{m}_1(t,x+\Delta x) = \mathsf{m}_1(t-\Delta t,x+\Delta x) &+ \frac{1}{2\lambda}(\mathsf{m}_2(t-\Delta t,x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x))\\ &+ \frac{1-s_3}{2\lambda^2}(\mathsf{m}_3(t-\Delta t,x)-2\mathsf{m}_3(t-\Delta t,x+\Delta x)+\mathsf{m}_3(t-\Delta t,x+2\Delta x))\\ &+ \frac{s_3}{2\lambda^2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))).\end{aligned}$$

Summing this two equations gives

$$\begin{aligned}\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x) &= \mathsf{m}_1(t-\Delta t,x-\Delta x)+\mathsf{m}_1(t-\Delta t,x+\Delta x)\\ &+ \frac{1}{2\lambda}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x))\\ &+ \frac{1-s_3}{2\lambda^2}(\mathsf{m}_3(t-\Delta t,x-2\Delta x)-2\mathsf{m}_3(t-\Delta t,x-\Delta x)+2\mathsf{m}_3(t-\Delta t,x)-2\mathsf{m}_3(t-\Delta t,x+\Delta x)+\mathsf{m}_3(t-\Delta t,x+2\Delta x))\\ &+ \frac{s_3}{2\lambda^2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-\Delta x))+2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))\\ &\quad -2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))),\end{aligned}$$

hence isolating the term in m_3 to eliminate in (7.12):

$$\begin{aligned}&\frac{1-s_3}{2}(\mathsf{m}_3(t-\Delta t,x-2\Delta x)-2\mathsf{m}_3(t-\Delta t,x-\Delta x)+2\mathsf{m}_3(t-\Delta t,x)-2\mathsf{m}_3(t-\Delta t,x\\ &+\Delta x)+\mathsf{m}_3(t-\Delta t,x+2\Delta x)) = \lambda^2(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))-\lambda^2(\mathsf{m}_1(t-\Delta t,x-\Delta x)+\mathsf{m}_1(t-\Delta t,x+\Delta x))\\ &-\frac{\lambda}{2}(\mathsf{m}_2(t-\Delta t,x-2\Delta x)-\mathsf{m}_2(t-\Delta t,x+2\Delta x))-\frac{s_3}{2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-2\Delta x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x-\Delta x))\\ &\quad +2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+\Delta x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t-\Delta t,x+2\Delta x))),\end{aligned}$$

which—plugged in (7.12)—gives

$$\begin{aligned}\mathsf{m}_3(t,x-\Delta x)-2\mathsf{m}_3(t,x)+\mathsf{m}_3(t,x+\Delta x) &= \lambda^2(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))\\ &-\lambda^2(\mathsf{m}_1(t-\Delta t,x-\Delta x)+\mathsf{m}_1(t-\Delta t,x+\Delta x))-\lambda(\mathsf{m}_2(t-\Delta t,x-\Delta x)-\mathsf{m}_2(t-\Delta t,x+\Delta x)).\end{aligned}\tag{7.13}$$

Inserting this equation into (7.6) renders the Finite Difference scheme for the conserved moment m_1:

$$\begin{aligned}\mathsf{m}_1(t+\Delta t,x) = \mathsf{m}_1(t,x) &+ \frac{1-s_3}{2}(\mathsf{m}_1(t,x-\Delta x)+\mathsf{m}_1(t,x+\Delta x))-\frac{1-s_3}{2}(\mathsf{m}_1(t-\Delta t,x-\Delta x)+\mathsf{m}_1(t-\Delta t,x+\Delta x))\\ &+ \frac{1}{2\lambda}(\mathsf{m}_2(t,x-\Delta x)-\mathsf{m}_2(t,x+\Delta x))-\frac{1-s_3}{2\lambda}(\mathsf{m}_2(t-\Delta t,x-\Delta x)-\mathsf{m}_2(t-\Delta t,x+\Delta x))\\ &+ \frac{s_3}{2\lambda^2}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-2m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x))+m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))).\end{aligned}\tag{7.14}$$

We recall the Finite Difference scheme for m_2:

$$\begin{aligned}\mathsf{m}_2(t+\Delta t,x) = \frac{1}{2}(2-s_3)(\mathsf{m}_2(t,x-\Delta x)+\mathsf{m}_2(t,x+\Delta x))-(1-s_3)\mathsf{m}_2(t-\Delta t,x)\\ +\frac{s_3}{2\lambda}(m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x-\Delta x))-m_3^{\mathrm{eq}}(\mathsf{m}_1(t,x+\Delta x))).\end{aligned}\tag{7.15}$$

To summarize, the key of these computations yielding (7.5) and (7.14)/(7.15) is to rewrite the scheme at different

time steps and at different points of the lattice and recombine the equations in order to get rid of the non-conserved moments. This is possible because the scheme is time-space invariant, so to speak, each point of the discrete lattice $\Delta t\mathbb{N} \times \Delta x \mathbb{Z}^d$ "sees" the same numerical scheme as the other points.

7.2 Elimination of "non-conserved" moments on ODEs

We now would like to perform the operations in the examples of Section 7.1 in full generality—that is—for any lattice Boltzmann scheme. To understand what is needed, the simple framework of linear systems of ODEs is quite helpful. On these systems, we can indeed perform an operation closely related to the elimination of the non-conserved moments.

Consider a system of linear ODEs of size $q \in \mathbb{N}^*$ made up of a matrix $\boldsymbol{A} \in \mathcal{M}_q(\mathbb{R})$:

$$\begin{cases} \boldsymbol{y}'(t) &= \boldsymbol{A}\boldsymbol{y}(t), \qquad t \geq 0, \\ \boldsymbol{y}(0) &= \boldsymbol{y}^\circ \in \mathbb{R}^q. \end{cases} \tag{7.16}$$

We could equally take the matrix with entries on any field, for example, $\boldsymbol{A} \in \mathcal{M}_q(\mathbb{C})$. This system is the analogue of the lattice Boltzmann scheme, in the sense that its state space is of dimension q and it features a first-order time derivative, as lattice Boltzmann scheme are one-step schemes. Two operations can be devised to and from (7.16) as in what follows.

- Transforming a single ODE of high order

$$\sum_{k=0}^{q} c_k y_1^{(k)}(t) = 0 \tag{7.17}$$

with $c_q = 1$ into a system of first order equations like (7.16) by considering $\boldsymbol{A}$ to be the companion matrix of the polynomial associated with equation (7.17), namely

$$\boldsymbol{A} = \begin{bmatrix} -c_{q-1} & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -c_2 & 0 & \cdots & 1 & 0 \\ -c_1 & 0 & \cdots & 0 & 1 \\ -c_0 & 0 & \cdots & 0 & 0 \end{bmatrix},$$

is a current practice, which unsurprisingly makes the problem more handy from the computational standpoint, *i.e.* to apply a numerical scheme to perform its time integration.

Example 7.2.1. *Consider the model for the one-dimensional linear harmonic oscillator $y_1'' + \frac{\kappa}{m} y_1 = 0$ for a mass $m > 0$ linked to a spring of constant $\kappa \geq 0$. The associated system reads*

$$\begin{bmatrix} y_1' \\ y_2' \end{bmatrix}(t) = \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{\kappa}{m} & 0 \end{bmatrix}}_{=\boldsymbol{A}} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}(t).$$

- Though being the analogous of what we aim at doing of lattice Boltzmann schemes, the other way around, namely passing from a system of first order (7.16) to a single equation of higher order (7.17), seems to be seldom considered. We proceed as in [Cull et al., 2005]: iterating, we have that $\boldsymbol{y}^{(k)} = \boldsymbol{A}^k \boldsymbol{y}$ for $k \in [\![0, q]\!]$. Let $(c_k)_{k \in [\![0,q]\!]} \subset \mathbb{R}$ be $q+1$ reals coefficients. We then form

$$\sum_{k=0}^{q} c_k \boldsymbol{y}^{(k)} = \left(\sum_{k=0}^{q} c_k \boldsymbol{A}^k \right) \boldsymbol{y}. \tag{7.18}$$

Taking $(c_k)_{k\in[\![0,q]\!]}$ to be the coefficients of the characteristic polynomial of $\boldsymbol{A}$, namely $\det(X\boldsymbol{I}-\boldsymbol{A}) = \sum_{k=0}^{k=q} c_k X^k$, by virtue of the Cayley-Hamilton theorem, the right hand side of (7.18) vanishes. We thus deduce the corresponding equation on the first variable y_1 (playing the role of the conserved moment in the lattice Boltzmann scheme), given by

$$\begin{cases} \sum_{k=0}^{k=q} c_k y_1^{(k)}(t) = 0, & t>0, \\ y_1(0) = (\boldsymbol{A}^0 \boldsymbol{y}^\circ)_1, \\ \vdots \\ y_1^{(q-1)}(0) = (\boldsymbol{A}^{q-1}\boldsymbol{y}^\circ)_1. \end{cases} \tag{7.19}$$

This provides a systematic way of performing the transformation without having to rely on hand computations and substitutions. It therefore tells us that we have to recast and understand the lattice Boltzmann schemes under a form which is suitable to apply a generalization of the Cayley-Hamilton theorem.

Example 7.2.2. *To give an example, consider the matrix*

$$\boldsymbol{A}_I = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 2 & 0 \end{bmatrix}, \qquad \textit{with} \qquad \det(X\boldsymbol{I}-\boldsymbol{A}_I) = X^3 - 3X^2 - 2X + 1.$$

The corresponding ODE of higher order (7.17) on the first variable y_1 therefore reads $y_1''' - 3y_1'' - 2y_1' + y_1 = 0$.

7.3 The Cayley-Hamilton theorem on commutative rings

Commutative rings are one general algebraic structure on which the Cayley-Hamilton theorem holds. It shall turn out that we can construct a commutative ring to represent our lattice Boltzmann schemes. We here present the Cayley-Hamilton theorem for commutative rings and an algorithm which allows to compute the characteristic polynomial quite rapidly.

Let $\mathcal{R}$ be a generic commutative ring. We shall here consider matrices with entries belonging to $\mathcal{R}$ [Brewer et al., 1986, Dummit and Foote, 2004] instead of fields such as $\mathbb{R}$ and $\mathbb{C}$. Still, the characteristic polynomial can be defined in a similar way:

Definition 7.3.1: Characteristic polynomial

Let $\mathcal{R}$ be a commutative ring and $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$. The characteristic polynomial of $\mathbf{C}$ is defined by $\det(X\boldsymbol{I} - \mathbf{C}) \in \mathcal{R}[X]$.

Algorithm 5 Faddeev-Leverrier algorithm for the computation of the characteristic polynomial of a square matrix on a commutative ring $\mathcal{R}$.

Input: $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$.
Set $\mathbf{D} = \mathbf{C}$
for $k \in [\![1, q]\!]$ **do**
  if $k > 1$ **then**
    Compute $\mathbf{D} = \mathbf{C}(\mathbf{D} + \mathsf{c}_{q-k+1}\boldsymbol{I})$
  end if
  Compute $\mathsf{c}_{q-k} = -\frac{\mathrm{tr}(\mathbf{D})}{k}$
end for
Output: the coefficients $(\mathsf{c}_k)_{k\in[\![0,q]\!]} \subset \mathcal{R}$ of the characteristic polynomial $\det(X\boldsymbol{I} - \mathbf{C}) = \sum_{k=0}^{k=q} \mathsf{c}_k X^k$.

The naive computation of the characteristic polynomial $\det(X\boldsymbol{I} - \mathbf{C})$ using its Definition 7.3.1 via the determinant could be computationally expensive, especially when dealing with symbolic computations like it shall be in our case. For this reason, we employ the Faddeev-Leverrier algorithm [Hou, 1998] which is of polynomial complexity, generally lower than that of the pivot method. The process is detailed in Algorithm 5 and only uses matrix-matrix multiplications and the computation of the trace, denoted by $\mathrm{tr}(\cdot)$.

Remark 7.3.1. *The algorithm proposed by [Fučík and Straka, 2021] heavily relies on the computation of the traces of matrices and therefore looks quite similar to Algorithm 5. We hence conjecture that their proof concerning the fact that their algorithm terminates within a finite number of steps is an alternative proof of the Cayley-Hamilton Theorem 7.3.1 that we are going to state in a moment.*

As previously pointed out, the Cayley-Hamilton theorem for matrices over a commutative ring [Brewer et al., 1986], is a central result to be used with lattice Boltzmann schemes in order to eliminate the non-conserved moments. It generalizes the same result holding for matrices on fields, utilized in Section 7.2.

Theorem 7.3.1: Cayley-Hamilton

Let $\mathcal{R}$ be a commutative ring and $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$. Then $\det(X\boldsymbol{I} - \mathbf{C})$ is a monic polynomial in the ring $\mathcal{R}[X]$ in the indeterminate X in the form

$$\det(X\boldsymbol{I} - \mathbf{C}) = X^q + \mathsf{c}_{q-1}X^{q-1} + \cdots + \mathsf{c}_1 X^1 + \mathsf{c}_0,$$

with $(\mathsf{c}_k)_{k\in[\![0,q]\!]} \subset \mathcal{R}$ such that

$$\mathbf{C}^q + \mathsf{c}_{q-1}\mathbf{C}^{q-1} + \cdots + \mathsf{c}_1\mathbf{C} + \mathsf{c}_0\boldsymbol{I} = \mathbf{0}.$$

This result states that any square matrix with entries in a commutative ring verifies its characteristic equation. Having characterized the algebraic structure that allows to obtain a quite general form of the Cayley-Hamilton Theorem 7.3.1, we now turn to a specific commutative ring $\mathcal{R}$.

7.4 Construction of a suitable commutative ring

We construct a commutative ring to encode the lattice Boltzmann scheme and then apply Theorem 7.3.1. We start by presenting the original way of conceiving the ring. Then, we observe that there are different points of view that can be adopted to interpret it. Here, the time variable does not play any role thus it is not listed.

We start by the following.

Definition 7.4.1: Lattice functions and linear maps on them

We define the space of lattice functions

$$\mathsf{F} := \{\mathsf{f} \text{ such that } \mathsf{f} : \Delta x\mathbb{Z}^d \to \mathbb{R}\}.$$

Moreover, we consider $\mathcal{L}(\mathsf{F},\mathsf{F})$, the space of linear maps from F to F.

We also observe that $\mathcal{L}(\mathsf{F},\mathsf{F})$ has the following property.

Proposition 7.4.1

$(\mathcal{L}(\mathsf{F},\mathsf{F}), +, \circ)$, where $+$ is the sum and $\circ$ the composition of linear maps, is a commutative ring.

We continue by recalling that the stream phase (1.4) implies discrete velocities $(\boldsymbol{c}_j)_{j\in[\![1,q]\!]} \subset \mathbb{Z}^d$. The idea is then to associate an operator to each of them.

Definition 7.4.2: Shift operators in space

Let $\boldsymbol{z} \in \mathbb{Z}^d$. Then the associated shift operator on the lattice $\Delta x\mathbb{Z}^d$, denoted $\mathsf{t}_{\boldsymbol{z}}$, is defined in the following way. Let $\mathsf{f} \in \mathsf{F}$ be any function defined on the space lattice, then the action of $\mathsf{t}_{\boldsymbol{z}}$ is given by

$$(\mathsf{t}_{\boldsymbol{z}}\mathsf{f})(\boldsymbol{x}) = \mathsf{f}(\boldsymbol{x} - \boldsymbol{z}\Delta x), \qquad \forall \boldsymbol{x} \in \Delta x\mathbb{Z}^d.$$

We also introduce the set of shift operators $\mathsf{T} := \{\mathsf{t}_{\boldsymbol{z}} : \boldsymbol{z} \in \mathbb{Z}^d\} \cong \mathbb{Z}^d$, where $\cong$ indicates that two structures are isomorphic.

The shift operators yield information sought in the upwind direction with respect to the considered velocity. The action of the composition of $\mathcal{L}(\mathsf{F},\mathsf{F})$ between shifts is directly inherited from the sum in $\mathbb{Z}^d$:

Lemma 7.4.1: Product of shift operators in space

The composition $\circ$ of $\mathcal{L}(\mathsf{F},\mathsf{F})$ is internal to $\mathsf{T} \subset \mathcal{L}(\mathsf{F},\mathsf{F})$, indeed

$$\mathsf{t}_{\boldsymbol{z}} \circ \mathsf{t}_{\boldsymbol{w}} = \mathsf{t}_{\boldsymbol{z}+\boldsymbol{w}}, \qquad \forall \boldsymbol{z}, \boldsymbol{w} \in \mathbb{Z}^d.$$

Henceforth, the product $\circ$ is understood whenever no ambiguity is possible. This operation provides an algebraic structure to the shifts, directly inherited from that of $\mathbb{Z}^d$.

Proposition 7.4.2: Group of shift operators in space

$(\mathsf{T},\circ)$ forms a commutative (Abelian) group.

Proof. Thanks to Proposition 7.4.1, the property follows from the fact that $\circ$ is internal to T. Alternatively, we have constructed an isomorphism between $(\mathsf{T},\circ)$ and $(\mathbb{Z}^d,+)$. The latter is an Abelian group, which achieves the proof. □

Remark 7.4.1 (Generators of T). *Upon introducing the generating displacements along each Cartesian axis, given by* $\mathsf{x}_\alpha = \mathsf{t}_{\boldsymbol{e}_\alpha}$ *for any* $\alpha \in [\![1,d]\!]$, *where* $\boldsymbol{e}_\alpha$ *is the* α*-th vector of the canonical basis, we observe that there is only "one movement" in each direction which generated the shifts* T*:*

$$\mathsf{T} = \langle \{\mathsf{x}_1, \dots, \mathsf{x}_d\} \rangle,$$

where $\langle\cdot\rangle$ *is the customary notation for the generating set of a group. We remark that the fact that* T *is finitely-generated since* d *is finite—though interesting—is not mandatory to state the results to come. The statement from Proposition 7.4.2 is finally what one really needs.*

If the lattice Boltzmann schemes were made only of the stream phase (1.4), the algebraic structure $(\mathsf{T},\circ)$ would be enough to describe them. Still, since also the collision phase (1.1) has to be taken into account, we have to add one more binary operation, which is non-internal to T. This yields the cornerstone of Chapter 7, namely the set of Finite Difference operators in space, which are finite combinations of weighted shifts operators *via* a sum. It is defined as follows, see [Milies et al., 2002, Chapter 3].

Definition 7.4.3: Finite Difference operators in space

The set of Finite Difference operators on the space lattice $\Delta x\,\mathbb{Z}^d$ is defined as

$$\mathsf{D} := \mathbb{R}\mathsf{T} = \left\{ \sum_{\mathsf{t}\in\mathsf{T}} \alpha_\mathsf{t}\mathsf{t}, \quad \text{where} \quad \{\alpha_\mathsf{t}\}_{\mathsf{t}\in\mathsf{T}} \subset \mathbb{R} \text{ is compactly supported} \right\} \subset \mathcal{L}(\mathsf{F},\mathsf{F}), \tag{7.20}$$

the group ring (or group algebra) of T over $\mathbb{R}$. The "sum" $+ : \mathsf{D}\times\mathsf{D} \to \mathsf{D}$ and the "product" $\circ : \mathsf{D}\times\mathsf{D} \to \mathsf{D}$ of two elements are defined by

$$\left(\sum_{\mathsf{t}\in\mathsf{T}} \alpha_\mathsf{t}\mathsf{t}\right) + \left(\sum_{\mathsf{h}\in\mathsf{T}} \beta_\mathsf{h}\mathsf{h}\right) = \sum_{\mathsf{t}\in\mathsf{T}} (\alpha_\mathsf{t}+\beta_\mathsf{t})\mathsf{t}, \qquad \left(\sum_{\mathsf{t}\in\mathsf{T}} \alpha_\mathsf{t}\mathsf{t}\right) \circ \left(\sum_{\mathsf{h}\in\mathsf{T}} \beta_\mathsf{h}\mathsf{h}\right) = \sum_{\mathsf{t},\mathsf{h}\in\mathsf{T}} (\alpha_\mathsf{t}+\beta_\mathsf{h})(\mathsf{t}\circ\mathsf{h}).$$

Furthermore, the product of $\sigma \in \mathbb{R}$ with an element of D is given by

$$\sigma\left(\sum_{\mathsf{t}\in\mathsf{T}} \alpha_\mathsf{t}\mathsf{t}\right) = \sum_{\mathsf{t}\in\mathsf{T}} (\sigma\alpha_\mathsf{t})\mathsf{t}.$$

With the two binary operations, D behaves like $\mathbb{Z}$ and almost like $\mathbb{R}$ and $\mathbb{C}$ as well, as stated by the following result, see [Milies et al., 2002].

Proposition 7.4.3: Ring of Finite Difference operators in space

$(\mathsf{D}, +, \circ)$ is a commutative ring.

Hence, we see that we can apply Theorem 7.3.1 with ring $\mathcal{R} = \mathsf{D}$. Before going on with this idea, let us stress some features of D.

Remark 7.4.2 ($(\mathsf{D}, +, \circ)$ is not a field)**.** *Observe that* $(\mathsf{D}, +, \circ)$ *is not a field, since not every element* D *has multiplicative inverse. As an example, consider the centered approximation of the first derivative for* $d = 1$*, which is* $(\mathsf{t}_{-1} - \mathsf{t}_1)/(2\Delta x)$ *and refer to the notion of indefinite sum in the calculus of Finite Difference [Milne-Thomson, 1933, Miller, 1960]. The elements having inverse are called "units" and divide all the other elements for* D*. It can be easily seen that the units are the product of a non-zero real number and a shift in* T*. Indeed* $(\alpha \mathsf{t}_{\boldsymbol{z}})^{-1} = (1/\alpha)\mathsf{t}_{-\boldsymbol{z}}$ *for any* $\alpha \in \mathbb{R} \setminus \{0\}$ *and* $\boldsymbol{z} \in \mathbb{Z}^d$*. Observe also that the reals* $\mathbb{R}$ *can be identified with the sub-ring* $\mathbb{R} \cong \{\alpha \mathsf{t}_{\mathbf{0}} \,:\, \alpha \in \mathbb{R}\}$*.*

Remark 7.4.3 (D is more than a commutative ring)**.** *By Definition 7.4.3, since* $\mathbb{R}$ *is commutative,* D *can be also seen as an algebra over* $\mathbb{R}$*. It is also an Hopf algebra over* $\mathbb{R}$ *since* $\mathbb{R}$ *is a field. Moreover,* D *can be viewed as a free module where the scalars belong to* $\mathbb{R}$ *and the basis is formed by the elements of the group* T*. It can also be easily shown that* D *is a unique factorization domain (UFD), which encompass but are not limited to commutative rings.*

Remark 7.4.4 (Alternative constructions of D)**.** *Even if we have presented our way of constructing* D*, different derivations and interpretations are possible. Let us analyze some of them.*

- *Following [Cheng, 2003, Chapter 2], we could see functions on the lattice* $\Delta x \mathbb{Z}^d$ *as sequences and Finite Difference operators as sequences with compact support, whence the "compactly supported" in (7.20). By doing so, the product* $\circ$ *can be seen as a convolution between compactly supported sequences and the action of a Finite Difference operator on a function as the convolution of a finitely supported sequence with a generic sequence.*

- D *can be isomorphically identified with the ring of multivariate Laurent polynomials with real coefficients in the indeterminates* $\mathsf{x}_1, \dots, \mathsf{x}_d$*, that is*

$$\mathsf{D} \cong \mathbb{R}[\mathsf{x}_1, \mathsf{x}_1^{-1}, \dots, \mathsf{x}_d, \mathsf{x}_d^{-1}],$$

  which shall be endowed with the standard sum and product of polynomials. This automatically implies that D *is a unique factorization domain. This identification can somehow be interpreted as the historical starting point of umbral calculus [Roman, 2005], also known as calculus of Finite Difference [Miller, 1960]: allow to interchange indices in sequences (operators or functions on the lattice* $\Delta x \mathbb{Z}^d$*) with exponents (in polynomials). In this case, for any* $\boldsymbol{z} \in \mathbb{Z}^d$*, we can see* $\mathsf{t}_{\boldsymbol{z}} = \mathbf{x}^{\boldsymbol{z}}$ *using the multi-index notation where* $\mathbf{x} = (\mathsf{x}_1, \dots, \mathsf{x}_d)$ *and thus* $\mathbf{x}^{\boldsymbol{z}} = \mathsf{x}_1^{z_1} \cdots \mathsf{x}_d^{z_d}$*.*

- *Finally, the same construction can be done by considering the discrete Fourier transform, using the standard product and sum in* $\mathbb{C}$*, see Section 7.7.*

It is useful to define the conjugate operator of an operator $\mathsf{d} = \sum_{\mathsf{t} \in \mathsf{T}} \alpha_{\mathsf{t}} \mathsf{t} \in \mathsf{D}$, denoted by $\overline{\mathsf{d}}$ as $\overline{\mathsf{d}} := \sum_{\mathsf{t} \in \mathsf{T}} \alpha_{\mathsf{t}} \mathsf{t}^{-1} \in \mathsf{D}$. By interpreting D as a ring of Laurent polynomials, that is $\mathsf{d} = \mathsf{d}(\mathbf{x})$, we obtain $\overline{\mathsf{d}} = \overline{\mathsf{d}}(\mathbf{x}) = \mathsf{d}(\mathbf{x}^{-1})$. When one uses the interpretation using the Fourier transform, one relies on the conjugation over complex numbers. The conjugate operator allows to define symmetric and anti-symmetric parts for any operator in D.

Definition 7.4.4: Symmetric and anti-symmetric parts

Let $\mathsf{d} \in \mathsf{D}$, then we define

$$\mathsf{S}(\mathsf{d}) = \frac{1}{2}(\mathsf{d} + \overline{\mathsf{d}}), \qquad \mathsf{A}(\mathsf{d}) = \frac{1}{2}(\mathsf{d} - \overline{\mathsf{d}}),$$

where $\mathsf{S}(\mathsf{d})$ is called symmetric part of d and $\mathsf{A}(\mathsf{d})$ is called anti-symmetric part of d.

As usual, the symmetric part is the conjugate of itself and the anti-symmetric is the anti-conjugate of itself.

7.5 Corresponding Finite Difference schemes

Using the commutative ring D and Theorem 7.3.1, we can now rewrite any lattice Boltzmann scheme as a multi-step Finite Difference scheme, in order to algebraically and exactly eliminate the non-conserved moments.

In order to rewrite the lattice Boltzmann in a form similar to (7.16), we can take advantage of the operators in T to rewrite the stream phase (1.4) as (from now on, we do not use parenthesis when an operator acts on a function)

$$\mathbf{f}(t+\Delta t,\boldsymbol{x}) = \operatorname{diag}(\mathsf{t}_{\boldsymbol{c}_1},\dots,\mathsf{t}_{\boldsymbol{c}_q})\mathbf{f}^\star(t,\boldsymbol{x}), \qquad t\in\Delta t\mathbb{N}, \quad \boldsymbol{x}\in\Delta x\mathbb{Z}^d.$$

Here, the matrix belongs to $\mathcal{M}_q(\mathsf{D})$ which forms a non-commutative ring (as for real matrices) under the usual operations. This can be recast on the moments [Dubois, 2022, Farag et al., 2021]—in a non-diagonal form—by introducing $\mathbf{T} := \boldsymbol{M}\operatorname{diag}(\mathsf{t}_{\boldsymbol{c}_1},\dots,\mathsf{t}_{\boldsymbol{c}_q})\boldsymbol{M}^{-1} \in \mathcal{M}_q(\mathsf{D})$, representing the stream phase on the moments. Merging with the collision phase (1.1), we obtain a monolithic scheme given by

$$\mathbf{m}(t+\Delta t,\boldsymbol{x}) = \mathbf{A}\mathbf{m}(t,\boldsymbol{x}) + \mathbf{B}\mathbf{m}^{\mathrm{eq}}(t,\boldsymbol{x}), \qquad t\in\Delta t\mathbb{N}, \quad \boldsymbol{x}\in\Delta x\mathbb{Z}^d, \tag{7.21}$$

where we have introduced $\mathbf{A} := \mathbf{T}(\boldsymbol{I}-\boldsymbol{S}) \in \mathcal{M}_q(\mathsf{D})$ and $\mathbf{B} := \mathbf{T}\boldsymbol{S} \in \mathcal{M}_q(\mathsf{D})$ and we employ the shorthand

$$\mathbf{m}^{\mathrm{eq}}(t,\boldsymbol{x}) = \boldsymbol{m}^{\mathrm{eq}}(\mathsf{m}_1(t,\boldsymbol{x}),\dots,\mathsf{m}_N(t,\boldsymbol{x})),$$

to indicate the moments at equilibrium evaluated on the current solution, in a particular on the conserved moments. In the rest of Chapter 7, the point of the lattice $\boldsymbol{x}\in\Delta x\mathbb{Z}^d$ shall generally be dropped for the sake of readability. We observe several things.

Remark 7.5.1 (Eigenvalues of $\mathbf{A}$)**.** *The operators $(\mathsf{t}_{\boldsymbol{c}_j})_{j\in[\![1,q]\!]} \subset \mathsf{T} \subset \mathsf{D}$ are the eigenvalues of the matrix $\mathbf{T}$. However, they are not the eigenvalues of the matrix $\mathbf{A}$ (or $\mathbf{B}$). Indeed, it is in general false that these eigenvalues belong to the ring of Finite Difference operators D. This is analogous to the concept of "pseudo-scheme", see [Strikwerda, 2004, Section 10.6], because these eigenvalues do not represent Finite Difference operators but essentially act like them. Their action is easily defined using the Fourier transform, see Section 7.7.*

Remark 7.5.2 (Control systems)**.** *It is interesting to interpret the lattice Boltzmann scheme under the form (7.21) as discrete-time linear control system with matrices on a commutative ring [Brewer et al., 1986]. This will be especially useful in Chapter 10. The moments $\mathbf{m}$ are the state of the system evolving through the matrix $\mathbf{A}$, whereas the equilibria are the control* via $\mathbf{B}$ *being a non-linear feedback observing only a part of the state, namely the conserved moments.*

Finally, observe that the only parts of (7.21) where the non-conserved moments $\mathsf{m}_{N+1},\dots,\mathsf{m}_q$ are present are the left hand side and the term multiplied by $\mathbf{A}$. The term with $\mathbf{B}$ is fine since it contains only terms depending on the conserved moments, which we finally aim at keeping.

Before proceeding, we showcase the matrices $\mathbf{T}$, $\mathbf{A}$ and $\mathbf{B}$ for the examples of Section 7.1 and more.

Example 7.5.1 ($\mathrm{D}_1\mathrm{Q}_2$ scheme)**.** *Concerning the example of Section 7.1.1, we obtain that*

$$\mathbf{T} = \begin{bmatrix} \mathsf{S}(\mathsf{x}_1) & \frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1) \\ \lambda\mathsf{A}(\mathsf{x}_1) & \mathsf{S}(\mathsf{x}_1) \end{bmatrix}, \qquad \mathbf{A} = \begin{bmatrix} (1-s_1)\mathsf{S}(\mathsf{x}_1) & \frac{1-s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) \\ (1-s_1)\lambda\mathsf{A}(\mathsf{x}_1) & (1-s_2)\mathsf{S}(\mathsf{x}_1) \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} s_1\mathsf{S}(\mathsf{x}_1) & \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) \\ s_1\lambda\mathsf{A}(\mathsf{x}_1) & s_2\mathsf{S}(\mathsf{x}_1) \end{bmatrix}.$$

Here, we recall that we have symmetric $\mathsf{S}(\mathsf{x}_1) = (\mathsf{x}_1+\mathsf{x}_1^{-1})/2$ and anti-symmetric part $\mathsf{A}(\mathsf{x}_1) = (\mathsf{x}_1-\mathsf{x}_1^{-1})/2$ of x_1.

Example 7.5.2 ($\mathrm{D}_1\mathrm{Q}_3$ scheme)**.** *Concerning the example of Section 7.1.2, we have*

$$\mathbf{T} = \begin{bmatrix} 1 & \frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{1}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1) \\ 0 & \mathsf{S}(\mathsf{x}_1) & \frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1) \\ 0 & \lambda\mathsf{A}(\mathsf{x}_1) & \mathsf{S}(\mathsf{x}_1) \end{bmatrix},$$

and $\mathbf{A}$ and $\mathbf{B}$ accordingly.

Example 7.5.3 (D_1Q_3 scheme). *Consider the scheme introduced in Section 1.5.2 with moment matrix $\boldsymbol{M}$ given by (1.6). Then we have*

$$\mathbf{T} = \begin{bmatrix} \frac{1}{3}(2\mathsf{S}(\mathsf{x}_1)+1) & \frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{1}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1) \\ \frac{2\lambda}{3}\mathsf{A}(\mathsf{x}_1) & \mathsf{S}(\mathsf{x}_1) & \frac{1}{3\lambda}\mathsf{A}(\mathsf{x}_1) \\ \frac{2\lambda^2}{3}(\mathsf{S}(\mathsf{x}_1)-1) & \lambda\mathsf{A}(\mathsf{x}_1) & \frac{1}{3}(\mathsf{S}(\mathsf{x}_1)+2) \end{bmatrix},$$

and $\mathbf{A}$ and $\mathbf{B}$ accordingly.

7.5.1 One conserved moment

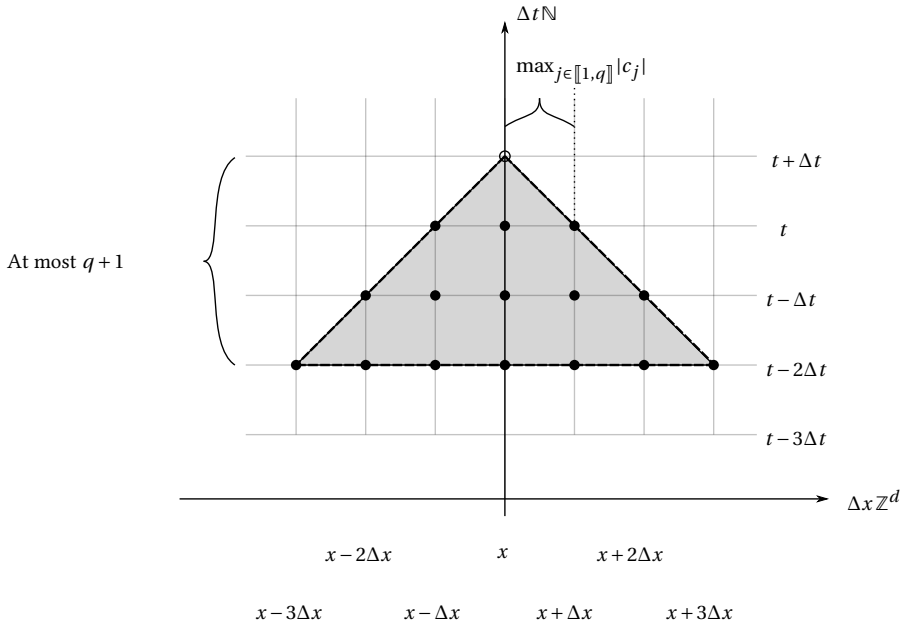


Figure 7.1: Maximal space-time domain of dependence of the corresponding Finite Difference scheme for $N=1$ (full black points inside the grey area) by virtue of Proposition 7.5.1 in the case of $d=1$. The maximal space-time slopes are determined by the maximal shift of the considered scheme whereas the number of involved time-steps is at most $q+1$.

We first analyze the case of one conserved moment, namely $N=1$, to keep the presentation as simple as possible and because it conveys the core idea. We shall eventually deal with $N>1$—which gives some additional difficulties—once the principles are established. The following result encompasses the findings from [Suga, 2010, Dellacherie, 2014].

Proposition 7.5.1: Corresponding Finite Difference scheme for $N=1$

Consider $N=1$. Then the lattice Boltzmann scheme given by (7.21) corresponds to a multi-step explicit Finite Difference scheme on the conserved moment m_1 under the form

$$\mathsf{m}_1(t+\Delta t,\boldsymbol{x}) = -\sum_{k=0}^{q-1} \mathsf{c}_k \mathsf{m}_1(t+(1-q+k)\Delta t,\boldsymbol{x}) + \left(\sum_{k=0}^{q-1}\left(\sum_{r=0}^{k} \mathsf{c}_{q+r-k}\mathbf{A}^r\right)\mathbf{B}\mathbf{m}^{\mathrm{eq}}(t-k\Delta t,\boldsymbol{x})\right)_1, \tag{7.22}$$

for all $t\in\Delta t[\![q-1,+\infty[\![$ and for all $\boldsymbol{x}\in\Delta x\mathbb{Z}^d$, where $(\mathsf{c}_k)_{k\in[\![0,q]\!]}\subset\mathsf{D}$ are the coefficients of $\det(X\boldsymbol{I}-\mathbf{A}) = \sum_{k=0}^{k=q}\mathsf{c}_k X^k$, the characteristic polynomial of $\mathbf{A}$.

This result means that the conserved moment satisfies an explicit multi-step Finite Difference scheme with at most q steps, thus involving $q+1$ discrete time instants, *cf.* Figure 7.1. The maximal size of spatial influence at each past time step can be deduced by looking at Algorithm 5, derived from the Newton's identities. It is interesting to observe that also the non-conserved moments satisfy a Finite Difference numerical scheme, see the following

proof. However, these schemes would depend on the conserved moment via the equilibria and are therefore not independent from the rest of the system.

Proof of Proposition 7.5.1. Consider any $k \in \mathbb{N}$ and let $t \in \Delta t[\![k-1, +\infty[\![$. Applying (7.21) recursively, we obtain

$$\mathbf{m}(t+\Delta t) = \mathbf{A}^k \mathbf{m}(t-(k-1)\Delta t) + \sum_{r=0}^{k-1} \mathbf{A}^r \mathbf{B}\mathbf{m}^{\mathrm{eq}}(t-r\Delta t).$$

We denote $\tilde{t} := t-(k-1)\Delta t$, obtaining

$$\mathbf{m}(\tilde{t}+k\Delta t) = \mathbf{A}^k \mathbf{m}(\tilde{t}) + \sum_{r=0}^{k-1} \mathbf{A}^r \mathbf{B}\mathbf{m}^{\mathrm{eq}}(\tilde{t}+(k-1-r)\Delta t),$$

which holds true, in particular, for any $k \in [\![0, q]\!]$. We can then consider the coefficients $(\mathsf{c}_k)_{k\in[\![0,q]\!]} \subset \mathsf{D}$ of the characteristic polynomial $\det(X\boldsymbol{I}-\mathbf{A}) = \sum_{k=0}^{k=q} \mathsf{c}_k X^k$ and write

$$\sum_{k=0}^{q} \mathsf{c}_k \mathbf{m}(\tilde{t}+k\Delta t) = \left(\sum_{k=0}^{q} \mathsf{c}_k \mathbf{A}^k\right)\mathbf{m}(\tilde{t}) + \sum_{k=0}^{q} \mathsf{c}_k \sum_{r=0}^{k-1} \mathbf{A}^r \mathbf{B}\mathbf{m}^{\mathrm{eq}}(\tilde{t}+(k-1-r)\Delta t). \tag{7.23}$$

Applying the Cayley-Hamilton Theorem 7.3.1, thanks to Proposition 7.4.3, gives $\sum_{k=0}^{k=q} \mathsf{c}_k \mathbf{A}^k = \mathbf{0}$. Using the monicity of the characteristic polynomial and changing time indices by setting $\tilde{t}+q\Delta t = t+\Delta t$ gives

$$\mathbf{m}(t+\Delta t) = -\sum_{k=0}^{q-1} \mathsf{c}_k \mathbf{m}(t+(1-q+k)\Delta t) + \sum_{k=0}^{q} \mathsf{c}_k \sum_{r=0}^{k-1} \mathbf{A}^r \mathbf{B}\mathbf{m}^{\mathrm{eq}}(t+(k-q-r)\Delta t).$$

Observe that the last sum can start from $k=1$. Performing a change of indices in the last double sum yields

$$\mathbf{m}(t+\Delta t) = -\sum_{k=0}^{q-1} \mathsf{c}_k \mathbf{m}(t+(1-q+k)\Delta t) + \sum_{k=0}^{q-1}\left(\sum_{r=0}^{k} \mathsf{c}_{q+r-k} \mathbf{A}^r\right)\mathbf{B}\mathbf{m}^{\mathrm{eq}}(t-k\Delta t). \tag{7.24}$$

The proof is achieved by selecting the first component in (7.24). □

Comparing (7.21) and (7.22)/(7.24) and looking at the proof of Proposition 7.5.1, we notice that we have somehow performed a sort of "diagonalization" of the matrix $\mathbf{A}$ to obtain something diagonal in the moments $\mathbf{m}$. We talk about a "sort of" because we have observed—*cf.* Remark 7.5.1—that the eigenvalues of $\mathbf{A}$ do not generally belong to D and there is no notion of change of basis in this framework.

Remark 7.5.3 (The fate of $\mathsf{m}_2, \dots, \mathsf{m}_q$)**.** *Remark that the non-conserved moments $\mathsf{m}_2, \dots, \mathsf{m}_q$ are no longer defined and they cannot be recovered from (7.22), because we have selected the first line in (7.24). Still, there is a residual "shadow" of their presence, namely the multi-step nature of the Finite Difference scheme (7.22).*

Remark 7.5.4 (Meaning of "corresponding")**.** *The word "corresponding" in Proposition 7.5.1 means that the original lattice Boltzmann scheme and the corresponding Finite Difference scheme issue the same discrete dynamics of the conserved moment m_1. This means that if one takes the initialization into account,* cf. *Chapter 10, the solutions will always be equal up to machine precision.*

Example 7.5.4 ($\mathrm{D}_1\mathrm{Q}_2$ scheme)**.** *Coming back to Example 7.5.1, we have that*

$$\det(X\boldsymbol{I}-\mathbf{A}) = X^2 - (2-s_1-s_2)\mathsf{S}(\mathsf{x}_1)X + (1-s_1)(1-s_2)\underbrace{(\mathsf{S}(\mathsf{x}_1)^2 - \mathsf{A}(\mathsf{x}_1)^2)}_{=1},$$

$$\textit{for} \quad k=0, \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \boldsymbol{e}_1^{\mathsf{t}}\boldsymbol{I}\underbrace{\begin{bmatrix} s_1\mathsf{S}(\mathsf{x}_1) & \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) \\ s_1\lambda\mathsf{A}(\mathsf{x}_1) & s_2\mathsf{S}(\mathsf{x}_1)\end{bmatrix}}_{=\mathbf{B}} = \left(s_1\mathsf{S}(\mathsf{x}_1), \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1)\right),$$

$$\text{for} \quad k=1, \qquad \left(\sum_{r=0}^{k} c_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \boldsymbol{e}_1^{\mathsf{t}}\Bigg(\underbrace{-(2-s_1-s_2)\mathsf{S}(\mathsf{x}_1)}_{=c_1}\boldsymbol{I} + \underbrace{\begin{bmatrix}(1-s_1)\mathsf{S}(\mathsf{x}_1) & \frac{1-s_2}{\lambda}\mathsf{A}(\mathsf{x}_1)\\ (1-s_1)\lambda\mathsf{A}(\mathsf{x}_1) & (1-s_2)\mathsf{S}(\mathsf{x}_1)\end{bmatrix}}_{=\mathbf{A}}\Bigg) \times \underbrace{\begin{bmatrix}s_1\mathsf{S}(\mathsf{x}_1) & \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1)\\ s_1\lambda\mathsf{A}(\mathsf{x}_1) & s_2\mathsf{S}(\mathsf{x}_1)+\end{bmatrix}}_{=\mathbf{B}} = (s_1(s_2-1),0).$$

yielding, thanks to (1.3), the corresponding Finite Difference scheme by (7.22)

$$\mathsf{m}_1(t+\Delta t) = (2-s_2)\mathsf{S}(\mathsf{x}_1)\mathsf{m}_1(t) - (1-s_2)\mathsf{m}_1(t-\Delta t) + \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2^{\mathrm{eq}}(t),$$

which is exactly the previously computed (7.5), which has been obtained by direct computations.

Example 7.5.5 ($\mathrm{D_1Q_3}$ with $N=1$)**.** *Consider the scheme in Example 7.5.3. As the reader might have noticed from the previous Example 7.5.4, the terms containing s_1, which does not influence the lattice Boltzmann scheme, simplify when computing the corresponding Finite Difference scheme, as we shall rigorously demonstrate in Section 8.2. It is therefore harmless but useful to have less terms when writing down the equation, to assume that $s_1=0$. With this:*

$$\det(X\boldsymbol{I}-\mathbf{A}) = X^3 + \Big(-2\mathsf{S}(\mathsf{x}_1) - 1 + s_2\mathsf{S}(\mathsf{x}_1) + \frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\Big)X^2 + \Big(1+2\mathsf{S}(\mathsf{x}_1) - s_2(\mathsf{S}(\mathsf{x}_1)+1) - \frac{5s_3}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{5}\Big) + \frac{2s_2s_3}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{2}\Big)\Big)X - (1-s_2)(1-s_3),$$

and

$$\text{for} \quad k=0, \qquad \left(\sum_{r=0}^{k} c_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \boldsymbol{e}_1^{\mathsf{t}}\boldsymbol{I}\underbrace{\begin{bmatrix}0 & \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\\ 0 & s_2\mathsf{S}(\mathsf{x}_1) & \frac{s_3}{3\lambda}\mathsf{A}(\mathsf{x}_1)\\ 0 & \lambda s_2\mathsf{A}(\mathsf{x}_1) & \frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\end{bmatrix}}_{=\mathbf{B}} = \Big(0, \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1), \frac{s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\Big),$$

$$\text{for} \quad k=1, \qquad \left(\sum_{r=0}^{k} c_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \boldsymbol{e}_1^{\mathsf{t}}\Bigg(\Big(-2\mathsf{S}(\mathsf{x}_1) - 1 + s_2\mathsf{S}(\mathsf{x}_1) + \frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\Big)\boldsymbol{I} + \underbrace{\begin{bmatrix}\frac{1}{3}(2\mathsf{S}(\mathsf{x}_1)+1) & \frac{1-s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{1-s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\\ \frac{2\lambda}{3}\mathsf{A}(\mathsf{x}_1) & (1-s_2)\mathsf{S}(\mathsf{x}_1) & \frac{1-s_3}{3\lambda}\mathsf{A}(\mathsf{x}_1)\\ \frac{2\lambda^2}{3}(\mathsf{S}(\mathsf{x}_1)-1) & \lambda(1-s_2)\mathsf{A}(\mathsf{x}_1) & \frac{1-s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\end{bmatrix}}_{=\mathbf{A}}\Bigg) \times \underbrace{\begin{bmatrix}0 & \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\\ 0 & s_2\mathsf{S}(\mathsf{x}_1) & \frac{s_3}{3\lambda}\mathsf{A}(\mathsf{x}_1)\\ 0 & \lambda s_2\mathsf{A}(\mathsf{x}_1) & \frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\end{bmatrix}}_{=\mathbf{B}} = \Big(0, \frac{s_2(s_3-1)}{\lambda}\mathsf{A}(\mathsf{x}_1), \frac{(1-s_2)s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\Big),$$

$$\text{for } k=2, \quad \left(\sum_{r=0}^{k} c_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \boldsymbol{e}_1^{\mathsf{t}}\Bigg(\underbrace{\Big(1+2\mathsf{S}(\mathsf{x}_1) - s_2(\mathsf{S}(\mathsf{x}_1)+1) - \frac{5s_3}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{5}\Big) + \frac{2s_2s_3}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{2}\Big)\Big)}_{=c_1}\boldsymbol{I} + \underbrace{\Big(-2\mathsf{S}(\mathsf{x}_1) - 1 + s_2\mathsf{S}(\mathsf{x}_1) + \frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\Big)}_{=c_2}\underbrace{\begin{bmatrix}\frac{1}{3}(2\mathsf{S}(\mathsf{x}_1)+1) & \frac{1-s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{1-s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\\ \frac{2\lambda}{3}\mathsf{A}(\mathsf{x}_1) & (1-s_2)\mathsf{S}(\mathsf{x}_1) & \frac{1-s_3}{3\lambda}\mathsf{A}(\mathsf{x}_1)\\ \frac{2\lambda^2}{3}(\mathsf{S}(\mathsf{x}_1)-1) & \lambda(1-s_2)\mathsf{A}(\mathsf{x}_1) & \frac{1-s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\end{bmatrix}}_{=\mathbf{A}}\Bigg)$$

$$+\mathbf{A}^2\Big)\underbrace{\begin{bmatrix}0 & \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) & \frac{s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\\ 0 & s_2\mathsf{S}(\mathsf{x}_1) & \frac{s_3}{3\lambda}\mathsf{A}(\mathsf{x}_1)\\ 0 & \lambda s_2\mathsf{A}(\mathsf{x}_1) & \frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\end{bmatrix}}_{=\mathbf{B}} = (0,0,0),$$

where the terms in $\mathbf{A}^2$ *are quite involved to be written in the previous formula. They read, for example*

$$(\mathbf{A}^2)_{11} = \frac{1}{3}(1+2\mathsf{S}(\mathsf{x}_1^2)) + \frac{s_2}{3}(1-\mathsf{S}(\mathsf{x}_1^2)) + \frac{s_3}{9}(-3+4\mathsf{S}(\mathsf{x}_1)-\mathsf{S}(\mathsf{x}_1^2)),$$
$$(\mathbf{A}^2)_{12} = \frac{1}{\lambda}\Big(\mathsf{A}(\mathsf{x}_1^2) - \frac{3s_2}{2}\mathsf{A}(\mathsf{x}_1^2) + \frac{s_3}{6}(2\mathsf{A}(\mathsf{x}_1)-\mathsf{A}(\mathsf{x}_1^2)) + \frac{s_2^2}{2}\mathsf{A}(\mathsf{x}_1^2) + \frac{s_2s_3}{6}(-2\mathsf{A}(\mathsf{x}_1)+\mathsf{A}(\mathsf{x}_1^2))\Big),$$

and

$$(\mathbf{A}^2)_{13} = \frac{1}{3\lambda^2}\Big(-1+\mathsf{S}(\mathsf{x}_1^2) + \frac{s_2}{2}(1-\mathsf{S}(\mathsf{x}_1^2)) + \frac{s_3}{6}(9-2\mathsf{S}(\mathsf{x}_1)-7\mathsf{S}(\mathsf{x}^2)) + \frac{s_2s_3}{2}(-2+\mathsf{S}(\mathsf{x}_1^2)) + \frac{s_3^2}{6}(-3+2\mathsf{S}(\mathsf{x}_1)+\mathsf{S}(\mathsf{x}_1^2))\Big).$$

This boils down to the corresponding Finite Difference scheme which reads

$$\begin{aligned}\mathsf{m}_1(t+\Delta t) &= \Big(2\mathsf{S}(\mathsf{x}_1)+1-s_2\mathsf{S}(\mathsf{x}_1)-\frac{s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2)\Big)\mathsf{m}_1(t)\\ &+\Big(-1-2\mathsf{S}(\mathsf{x}_1)+s_2(\mathsf{S}(\mathsf{x}_1)+1)+\frac{5s_3}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{5}\Big)-\frac{2s_2s_3}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{2}\Big)\Big)\mathsf{m}_1(t-\Delta t)+(1-s_2)(1-s_3)\mathsf{m}_1(t-2\Delta t)\\ &+\frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2^{\mathrm{eq}}(t)+\frac{s_3(s_3-1)}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2^{\mathrm{eq}}(t-\Delta t)+\frac{s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t)+\frac{(1-s_2)s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t-\Delta t). \quad (7.25)\end{aligned}$$

Remark 7.5.5 (Memory occupation)**.** *Before dealing with several conserved moments, namely* $N>1$*, let us point out that compared to the original lattice Boltzmann scheme, where either* $\mathbf{f}\in\mathbb{R}^q$ *or* $\mathbf{m}\in\mathbb{R}^q$ *at time* t *has to be stored at each node of the mesh* $\Delta x\mathbb{Z}^d$ *to evolve the solution, if we consider the Finite Difference formulation (7.22), we store only* m_1 *but at discrete times* $t, t-\Delta t,\dots,t-(q-1)\Delta t$*. Moreover, if we take, in Example 7.5.3,* $s_3=1$*, the third moment relaxes at its equilibrium which totally determines it at each time step as function of the conserved moment. In this case, in the original lattice Boltzmann scheme, one could only store* m_1 *and* m_2 *at the previous time at each node of* $\Delta x\mathbb{Z}^d$*. In the same fashion, (7.25) would involve only* m_1 *at time* t *and* $t-\Delta t$*, hence again, two variables per node. This means that the Finite Difference formulation does not allow, in general, to save memory compared to the original lattice Boltzmann method.*

Remark 7.5.6 (Time-space dependent schemes)**.** *One could think of allowing* $\boldsymbol{M}=\boldsymbol{M}(t,\boldsymbol{x})$ *and/or* $\boldsymbol{S}=\boldsymbol{S}(t,\boldsymbol{x})$ *to depend on the time and space variables. This would imply to consider weights in Definition 7.4.3 made up of functions instead of real numbers. However in this case,* D *would no longer be commutative because the multiplication by a function does not commute with the shifts, see [Rota et al., 1973, "shift invariance"]. For example, take* $\boldsymbol{z}\in\mathbb{Z}^d$ *and a function* $\mathsf{g}:\Delta x\mathbb{Z}^d\to\mathbb{R}$*, then*

$$(\mathsf{t}_{\boldsymbol{z}}\circ(\mathsf{g}\mathsf{t}_{\mathbf{0}}))\mathsf{f}(\boldsymbol{x}) = \mathsf{g}(\boldsymbol{x}-\boldsymbol{z}\Delta x)\mathsf{f}(\boldsymbol{x}-\boldsymbol{z}\Delta x), \qquad ((\mathsf{g}\mathsf{t}_{\mathbf{0}})\circ\mathsf{t}_{\boldsymbol{z}})\mathsf{f}(\boldsymbol{x}) = \mathsf{g}(\boldsymbol{x})\mathsf{f}(\boldsymbol{x}-\boldsymbol{z}\Delta x), \qquad \forall\boldsymbol{x}\in\Delta x\mathbb{Z}^d,$$

for any function $\mathsf{f}:\Delta x\mathbb{Z}^d\to\mathbb{R}$*. The right hand sides are not equal in general, except when* g *is constant, which comes back to the setting on Definition 7.4.3.*

7.5.2 Several conserved moments

Let us consider the case of several conserved moments, that is, $N>1$. In this Section 7.5.2, we first observe on an example that it is not indicated to proceed exactly like for $N=1$, for several reasons. Then, we introduce a decomposition of the scheme matrix $\mathbf{A}$ in order to avoid these shortcomings by treating each conserved moment differently from the others. Finally, we use this new way of writing the scheme as for $N=1$, namely invoking the Cayley-Hamilton Theorem 7.3.1.

The first path that we might try to follow is to restart from (7.24) in the proof of Proposition 7.5.1 and to select the i-th line in this identity with $i\in[\![1,N]\!]$ spanning the conserved moments. Let us do this on the example of

Section 7.1.2.

Example 7.5.6 ($\mathrm{D_1Q_3}$ with $N=2$). *Consider the scheme in Section 7.1.2 for $N=2$ and let us focus on the first moment m_1, that is, consider $i=1$. We obtain*

$$\begin{aligned}\det(X\boldsymbol{I}-\mathbf{A}) = X^3 &+ (-1-2\mathsf{S}(\mathsf{x}_1)+s_1+s_2\mathsf{S}(\mathsf{x}_1)+s_3\mathsf{S}(\mathsf{x}_1))X^2\\ &+ (1+2\mathsf{S}(\mathsf{x}_1)-2s_1\mathsf{S}(\mathsf{x}_1)-s_2(\mathsf{S}(\mathsf{x}_1)+1)-s_3(\mathsf{S}(\mathsf{x}_1)+1)+s_1s_2\mathsf{S}(\mathsf{x}_1)+s_1s_3\mathsf{S}(\mathsf{x}_1)+s_2s_3)X\\ &- (1-s_1)(1-s_2)(1-s_3),\end{aligned}$$

and

$$\begin{aligned}&\textit{for}\quad k=0, \qquad \left(\sum_{r=0}^{k}\mathsf{c}_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \left(s_1, \frac{s_2}{\lambda}\mathsf{A}((\mathsf{x}_1)), \frac{s_3}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\right),\\ &\textit{for}\quad k=1, \qquad \left(\sum_{r=0}^{k}\mathsf{c}_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = \left(s_1(-2+s_2+s_3)\mathsf{S}(\mathsf{x}_1), \frac{s_2}{\lambda}(-1+s_3)\mathsf{A}(\mathsf{x}_1), \frac{s_3(1-s_2)}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\right),\\ &\textit{for}\quad k=2, \qquad \left(\sum_{r=0}^{k}\mathsf{c}_{q+r-k}\mathbf{A}^r\mathbf{B}\right)_{1,\cdot} = (s_1(1-s_2)(1-s_3),0,0).\end{aligned}$$

This would yield the Finite Difference scheme

$$\begin{aligned}\mathsf{m}_1(t+\Delta t) = (1+2\mathsf{S}(\mathsf{x}_1)-s_2\mathsf{S}(\mathsf{x}_1)-s_3\mathsf{S}(\mathsf{x}_1))\mathsf{m}_1(t)&\\ + (-1-2\mathsf{S}(\mathsf{x}_1)+s_2(\mathsf{S}(\mathsf{x}_1)+1)+s_3(\mathsf{S}(\mathsf{x}_1)+1)-s_2s_3)\mathsf{m}_1(t-\Delta t) + (1-s_2)(1-s_3)\mathsf{m}_1(t-2\Delta t)&\\ + \frac{s_2}{\lambda}\mathsf{A}((\mathsf{x}_1))\mathsf{m}_2(t) + \frac{s_2}{\lambda}(-1+s_3)\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2(t-\Delta t)&\\ + \frac{s_3}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t) + \frac{s_3(1-s_2)}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t-\Delta t).&\quad (7.26)\end{aligned}$$

Looking at (7.26), we notice the following issues:

1. The scheme (7.26) does not coincide with (7.14) found by hand computations, even when $s_2=0$. This demonstrates that this procedure does not yield the same results that one "manually" obtains by trying to eliminate one conserved moment at each time without touching the remaining ones.
2. The scheme (7.26) depends on s_2, which is the fictitious relaxation parameter for the second conserved moment m_2. We observed that for the original lattice Boltzmann scheme, thanks to (1.3), the scheme does not depend on the $s_1,\dots,s_N$ associated to the conserved moments. The issue is that—quite the opposite—the corresponding Finite Difference scheme found in this way depends on this choice.
3. Applying the scheme (7.26) to smooth functions $m_1=m_1(t,x)$ and $m_2=m_2(t,x)$ with $(t,x)\in\mathbb{R}\times\mathbb{R}$ instead of the lattice functions m_1 and m_2, we can study the consistency by performing Taylor expansions. Under acoustic scaling, that is, fixing the lattice velocity λ as $\Delta x\to 0$, this yields

$$s_2s_3(\partial_t m_1+\partial_x m_2)=O(\Delta x).$$

When $s_2\neq 0$ (again observe that we should not have this dependence), the scheme is consistent with the first equation in (2.51), which would allow to conclude that this scheme is suitable to simulate the kind of system investigated in Section 2.8.1.4. However, when $s_2=0$, which is often taken, see [Février, 2014], the expansions have to be carried further in Δx and we obtain

$$s_3(\partial_{tt}m_1-\partial_{xx}(m_3^{\mathrm{eq}}(m_1)))=O(\Delta x).$$

What is the origin of this equation with second-order time derivative? The equations with which the original lattice Boltzmann scheme is consistent, using the analysis by [Dubois, 2008] are

$$\partial_t m_1+\partial_x m_2=O(\Delta x), \qquad (7.27)$$

$$\partial_t m_2 + \partial_x(m_3^{\mathrm{eq}}(m_1)) = O(\Delta x). \tag{7.28}$$

Formally taking ∂_t(7.27) and inserting (7.28) to get rid of $\partial_{tx} m_2$, one obtains the equation $\partial_{tt} m_1 - \partial_{xx}(m_3^{\mathrm{eq}}(m_1)) = O(\Delta x)$. This is the wave equation associated with the original system (7.27)/(7.28) of conservation laws. However, we would like to obtain corresponding Finite Difference schemes which, regardless of the number of conserved moments N, are consistent with systems of PDEs which are first-order in time, like (7.27)/(7.28).

The previous remarks motivate to follow a different path. The idea is to do what we performed on Section 7.1.2, that is, select a conserved moment and consider the other conserved moments as "slave" variables as the equilibria have been until so far, because they imply variables that we eventually want to keep in the Finite Difference formulation. Otherwise said, we do not want the other conserved moments to participate to the elimination of the non-conserved ones by means of the characteristic polynomial of some matrix. In particular, we utilize different polynomials for different conserved moments to obtain the Finite Difference schemes. To formalize this concept, for any square matrix $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$ on a commutative ring $\mathcal{R}$, we consider $\mathbf{C}_I := (\sum_{i\in I} \boldsymbol{e}_i \otimes \boldsymbol{e}_i)\mathbf{C}(\sum_{i\in I} \boldsymbol{e}_i \otimes \boldsymbol{e}_i)$ for any $I \subset [\![1, q]\!]$, being the matrix obtained by $\mathbf{C}$ where only the rows and the columns of indices in I are kept and the remaining ones are set to zero. We also consider the matrix $\mathbf{C}[I] \in \mathcal{M}_{\#(I)}(\mathcal{R})$ obtained by keeping only the rows and the columns indexed by I. A useful Corollary of Theorem 7.3.1 is the following

Corollary 7.5.1

Let $\mathcal{R}$ be a commutative ring, $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$ and $I \subset [\![1, q]\!]$, then one has

$$\det(X\boldsymbol{I}_q - \mathbf{C}_I) = X^{q-\#(I)} \det(X\boldsymbol{I}_{\#(I)} - \mathbf{C}[I]).$$

Moreover, the polynomial $\det(X\boldsymbol{I}_{\#(I)} - \mathbf{C}[I])$ is annihilated by $\mathbf{C}_I$.

Proof. The first part of the Corollary comes from the Laplace formula for the determinant. The second one comes from Theorem 7.3.1 applied to $\mathbf{C}_I$. □

This means that the characteristic polynomial of $\mathbf{C}_I$ is directly linked to that of the smaller matrix $\mathbf{C}[I]$, which is thus faster to compute, and that the latter is an annihilator for the first matrix. Coming back to lattice Boltzmann schemes, for any conserved moment indexed by $i \in [\![1, N]\!]$, we introduce the matrices

$$\mathbf{A}_i = \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]}, \qquad \mathbf{A}_i^{\diamond} = \mathbf{A} - \mathbf{A}_i. \tag{7.29}$$

Notice that we have formed an additive decomposition $\mathbf{A} = \mathbf{A}_i + \mathbf{A}_i^{\diamond}$, which is of course not the only possible one, *cf.* Section 9.3. The idea is to "save" the moments other than the i-th into $\mathbf{A}_i^{\diamond}$ and exclude them from the computation of the characteristic polynomial, essentially like $\mathbf{B}$, which was excluded as well. With these notations, we have generated a family of problems of the same structure as (7.21) which read

$$\mathbf{m}(t + \Delta t) = \mathbf{A}_i \mathbf{m}(t) + \mathbf{A}_i^{\diamond} \mathbf{m}(t) + \mathbf{B}\mathbf{m}^{\mathrm{eq}}(t). \tag{7.30}$$

It is important to point out that the term in $\mathbf{A}_i\mathbf{m}$ inside (7.30) does not involve any conserved moment other than the i-th. Conversely, the term $\mathbf{A}_i^{\diamond}\mathbf{m}$ does not involve any function except the conserved moments other than the i-th. Then, the corresponding Finite Difference schemes come under the form stated by the following Proposition.

Proposition 7.5.2: Corresponding Finite Difference scheme for $N \geq 1$

Consider $N \geq 1$. Then the lattice Boltzmann scheme given by (7.21) or (7.30) corresponds to a family of multi-step explicit Finite Difference scheme on the conserved moment $\mathrm{m}_1, \ldots, \mathrm{m}_N$. This is, for any $i \in [\![1, N]\!]$

$$\mathrm{m}_i(t + \Delta t, \boldsymbol{x}) = -\sum_{k=0}^{q-N} \mathrm{c}_{i,k} \mathrm{m}_i(t + (N - q + k)\Delta t, \boldsymbol{x}) + \left(\sum_{k=0}^{q-N} \left(\sum_{r=0}^{k} \mathrm{c}_{i,1-N+q+r-k} \mathbf{A}_i^r \right) \mathbf{A}_i^{\diamond} \mathbf{m}(t - k\Delta t, \boldsymbol{x}) \right)_i \tag{7.31}$$

$$+\left(\sum_{k=0}^{q-N}\left(\sum_{r=0}^{k} \mathsf{c}_{i,1-N+q+r-k}\mathbf{A}_i^r\right)\mathbf{Bm}^{\mathrm{eq}}(t-k\Delta t,\boldsymbol{x})\right)_i,$$

for all $t \in \Delta t[\![q-N,+\infty[\![$ and for all $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$, where $\mathbf{A}_i := \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]}$ and $\mathbf{A}_i^\diamond := \mathbf{A} - \mathbf{A}_i$, with $(\mathsf{c}_{i,k})_{k\in[\![0,q+1-N]\!]} \subset \mathsf{D}$ are the coefficients of $\det(X\boldsymbol{I}-\mathbf{A}_i) = X^{N-1}\sum_{k=0}^{k=q+1-N}\mathsf{c}_{i,k}X^k$, the characteristic polynomial of $\mathbf{A}_i$.

Proof. The proof is analogous to the one of Proposition 7.5.1, taking advantage of Corollary 7.5.1. □

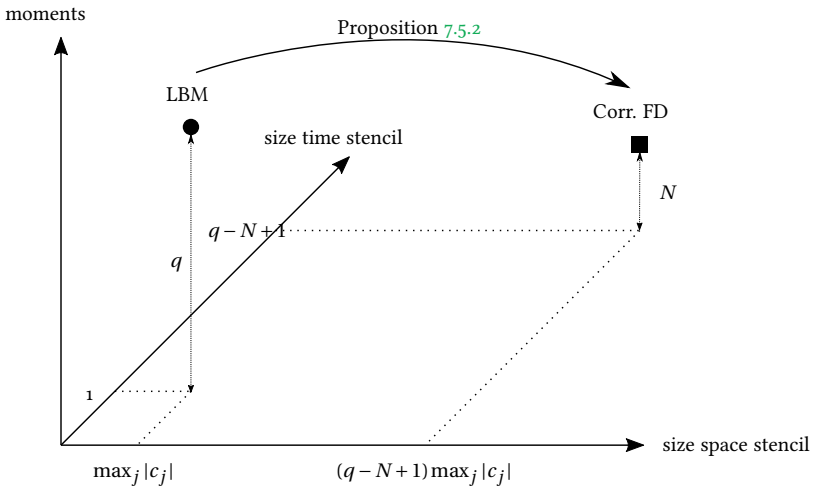


Figure 7.2: Comparison between lattice Boltzmann scheme (circle) and corresponding Finite Difference schemes (square) in terms of involved moments (respectively q and N), number of time steps (respectively 1 and $q-N+1$) and size of the maximal spatial stencil (respectively $\max_j|c_j|$ and $(q-N)\max_j|c_j|$).

Proposition 7.5.2 states that for each conserved moment, the corresponding Finite Difference scheme has at most $q-N+1$ steps, thus involves $q-N+2$ discrete times, see Figure 7.2. This result encompasses and generalizes Proposition 7.5.1. Observe that—at this stage—it is not totally clear whether the term featuring $\mathbf{A}_i^\diamond$ in (7.31) depends only on the conserved moments m_r with $r \in [\![1,N]\!] \smallsetminus \{i\}$ or mistakenly depends on the non-conserved moments $\mathsf{m}_{N+1},\dots,\mathsf{m}_q$ that we aim at eliminating. To see that this is indeed the case, namely this term depends only on the conserved moments, consider the example of $q=4$ and three conserved moments $N=3$. Then we would have, for the first moment

$$\mathbf{A}_1 = \begin{bmatrix} \star & 0 & 0 & \star \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \star & 0 & 0 & \star \end{bmatrix}, \qquad \mathbf{A}_1^\diamond = \begin{bmatrix} 0 & \star & \star & 0 \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ 0 & \star & \star & 0 \end{bmatrix},$$

where the starred $\star$ entries might be non-zero. We then obtain

$$\mathbf{A}_1^r = \underbrace{\begin{bmatrix} \star & 0 & 0 & \star \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \star & 0 & 0 & \star \end{bmatrix} \times\cdots\times \begin{bmatrix} \star & 0 & 0 & \star \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \star & 0 & 0 & \star \end{bmatrix}}_{r \text{ times}} = \begin{bmatrix} \star & 0 & 0 & \star \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \star & 0 & 0 & \star \end{bmatrix}.$$

Therefore, we have

$$\sum_{r=0}^{k} \mathsf{c}_{1,1-N+q+r-k}\mathbf{A}_1^r\mathbf{A}_1^{\diamond} = \begin{bmatrix} \star & 0 & 0 & \star \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \star & 0 & 0 & \star \end{bmatrix} \begin{bmatrix} 0 & \star & \star & 0 \\ \star & \star & \star & \star \\ \star & \star & \star & \star \\ 0 & \star & \star & 0 \end{bmatrix} = \begin{bmatrix} 0 & \star & \star & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \star & \star & 0 \end{bmatrix},$$

where we see that the first row—which we are eventually interested in—involves uniquely terms in the remaining conserved moments indexed by $i = 2,3$. This point shall discussed again in Section 9.3.

Example 7.5.7 ($\mathrm{D_1Q_3}$ with $N = 2$). *We come back to Example 7.5.6. Let us start by the first moment $i = 1$. We obtain*

$$\det(X\boldsymbol{I} - \mathbf{A}_1) = X^3 + (-1 - \mathsf{S}(\mathsf{x}_1) + s_1 + s_3\mathsf{S}(\mathsf{x}_1))X^2 + (\mathsf{S}(\mathsf{x}_1) - s_1\mathsf{S}(\mathsf{x}_1) - s_3\mathsf{S}(\mathsf{x}_1) + s_1s_3\mathsf{S}(\mathsf{x}_1))X,$$

where

$$\mathbf{A}_1 = \begin{bmatrix} \frac{1-s_1}{3}(2\mathsf{S}(\mathsf{x}_1)+1) & 0 & \frac{1-s_3}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1) \\ 0 & 0 & 0 \\ \frac{2\lambda^2(1-s_1)}{3}(\mathsf{S}(\mathsf{x}_1)-1) & 0 & \frac{1-s_3}{3}(\mathsf{S}(\mathsf{x}_1)+2) \end{bmatrix}, \qquad \mathbf{A}_1^{\diamond} = \begin{bmatrix} 0 & \frac{1-s_2}{\lambda}\mathsf{A}(\mathsf{x}_1) & 0 \\ \frac{2\lambda(1-s_1)}{3}\mathsf{A}(\mathsf{x}_1) & (1-s_2)\mathsf{S}(\mathsf{x}_1) & \frac{1-s_3}{3\lambda}\mathsf{A}(\mathsf{x}_1) \\ 0 & \lambda(1-s_2)\mathsf{A}(\mathsf{x}_1) & 0 \end{bmatrix},$$

along with

$$\text{for} \quad k = 0, \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{1,1-N+q+r-k}\mathbf{A}_1^r\mathbf{A}_1^{\diamond}\right)_{1,\cdot} = \left(0, \frac{1-s_2}{\lambda}\mathsf{A}(\mathsf{x}_1), 0\right),$$
$$\left(\sum_{r=0}^{k} \mathsf{c}_{1,1-N+q+r-k}\mathbf{A}_1^r\mathbf{B}\right)_{1,\cdot} = \left(s_1, \frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1), \frac{s_3}{\lambda}(\mathsf{S}(\mathsf{x}_1)-1)\right),$$

and

$$\text{for} \quad k = 1, \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{1,1-N+q+r-k}\mathbf{A}_1^r\mathbf{A}_1^{\diamond}\right)_{1,\cdot} = \left(0, \frac{-1+s_2+s_3-s_2s_3}{\lambda}\mathsf{A}(\mathsf{x}_1), 0\right),$$
$$\left(\sum_{r=0}^{k} \mathsf{c}_{1,1-N+q+r-k}\mathbf{A}_1^r\mathbf{B}\right)_{1,\cdot} = \left(s_1(s_3-1)\mathsf{S}(\mathsf{x}_1), \frac{s_2(-1+s_3)}{\lambda}\mathsf{A}(\mathsf{x}_1), 0\right),$$

yielding the corresponding Finite Difference scheme

$$\mathsf{m}_1(t+\Delta t) = (1+\mathsf{S}(\mathsf{x}_1) - s_3\mathsf{S}(\mathsf{x}_1))\mathsf{m}_1(t) - (1-s_3)\mathsf{S}(\mathsf{x}_1)\mathsf{m}_1(t-\Delta x) + \frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2(t) - \frac{1-s_3}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2(t-\Delta t) + \frac{s_3}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t),$$

which coincides with (7.14). Going on with the second conserved moment $i = 2$, one obtains

$$\det(X\boldsymbol{I} - \mathbf{A}_2) = X^3 + (-2\mathsf{S}(\mathsf{x}_1) + s_2\mathsf{S}(\mathsf{x}_1) + s_3\mathsf{S}(\mathsf{x}_1))X^2 + (1 - s_2 - s_3 + s_2s_3)X,$$

with

$$\text{for} \quad k = 0, \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{2,1-N+q+r-k}\mathbf{A}_2^r\mathbf{A}_2^{\diamond}\right)_{2,\cdot} = (0,0,0), \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{2,1-N+q+r-k}\mathbf{A}_2^r\mathbf{B}\right)_{2,\cdot} = \left(0, s_2\mathsf{S}(\mathsf{x}_1), \frac{s_3}{\lambda}\mathsf{A}(\mathsf{x}_1)\right),$$
$$\text{for} \quad k = 1, \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{2,1-N+q+r-k}\mathbf{A}_2^r\mathbf{A}_2^{\diamond}\right)_{2,\cdot} = (0,0,0), \qquad \left(\sum_{r=0}^{k} \mathsf{c}_{2,1-N+q+r-k}\mathbf{A}_2^r\mathbf{B}\right)_{2,\cdot} = (0, s_2(s_3-1), 0),$$

yielding the corresponding Finite Difference scheme

$$\mathsf{m}_2(t+\Delta t) = (2-s_3)\mathsf{S}(\mathsf{x}_1)\mathsf{m}_2(t) - (1-s_3)\mathsf{m}_2(t-\Delta t) + \frac{s_3}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_3^{\mathrm{eq}}(t),$$

which coincides with (7.15). We observe that, thanks to this procedure, we have eliminated the non-conserved moment

m_3, *obtaining schemes that do not depend on* s_1 *and* s_2*, as desired, and which are—under the scaling assumption that we have previously taken into account—consistent with the expected PDEs* (7.27)/(7.28).

7.6 Number of time-steps

Now that the main results Proposition 7.5.1 and Proposition 7.5.2 have been stated and proved, we can analyze and comment some particular cases. In particular, we discuss possible simplifications and schemes with less time steps in the process of elimination of non-conserved moments, since there exist cases in the literature where this takes place and we would like to account for them.

To illustrate some basic peculiarities and mechanisms that easily transpose to lattice Boltzmann schemes, we introduce the following matrices extending the discussion of Section 7.2:

$$\boldsymbol{A}_{\mathrm{I}} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 2 & 0 \end{bmatrix}, \qquad \boldsymbol{A}_{\mathrm{II}} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \qquad \boldsymbol{A}_{\mathrm{III}} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 2 & 0 \\ 1 & 2 & 1 \end{bmatrix}, \qquad \boldsymbol{A}_{\mathrm{IV}} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & -2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

- $\boldsymbol{A}_{\mathrm{I}}$ has been treated in Example 7.2.2 and it is a standard application of the result presented hitherto.
- For $\boldsymbol{A}_{\mathrm{II}}$, the characteristic polynomial is $\det(X\boldsymbol{I} - \boldsymbol{A}_{\mathrm{II}}) = X^3 - 5X^2 + 8X - 4$, corresponding to the higher order equation $y''' - 5y'' + 8y' - 4y = 0$. However, contrarily to $\boldsymbol{A}_{\mathrm{I}}$, the characteristic polynomial $\det(X\boldsymbol{I} - \boldsymbol{A}_{\mathrm{II}})$ does not coincide with the minimal polynomial $\mu_{\boldsymbol{A}_{\mathrm{II}}} = X^2 - 3X + 2$. Thus in this case, we could use the latter to obtain (7.19) having $y'' - 3y' + 2y = 0$ This phenomenon is studied in Section 7.6.1. It indicates that we can achieve a more compact corresponding ODE by using the annihilating polynomial of smallest degree on every variable. This does not change the core of the strategy.
- For $\boldsymbol{A}_{\mathrm{III}}$, we obtain $\det(X\boldsymbol{I} - \boldsymbol{A}_{\mathrm{III}}) = X^3 - 4X^2 + 4X - 1$, corresponding to $y''' - 4y'' + 4y' - y = 0$. However, by inspecting $\boldsymbol{A}_{\mathrm{III}}$, one notices that the first two equations do not depend on the last variable y_3. For this reason, we could have considered the matrix $\boldsymbol{A}_{\mathrm{III}}[\{1,2\}]$ obtained from $\boldsymbol{A}_{\mathrm{III}}$ by removing the last row and column. In this case $\det(X\boldsymbol{I}_2 - \boldsymbol{A}_{\mathrm{III}}[\{1,2\}]) = X^2 - 3X + 1$, corresponding to the equation $y'' - 3y' + y = 0$. This kind of situation for lattice Boltzmann schemes is investigated in Section 7.6.2. It is interesting to observe once more that $\det(X\boldsymbol{I}_2 - \boldsymbol{A}_{\mathrm{III}}[\{1,2\}])$ divides $\det(X\boldsymbol{I} - \boldsymbol{A}_{\mathrm{III}})$. This shows that an initial inspection of the matrix can yield a reduction of the size of the problem that can be achieved by a simple trimming operation, which eliminates some variable from the problem but treats the remaining ones as usual.
- Finally, consider $\boldsymbol{A}_{\mathrm{IV}}$. In this case the characteristic polynomial and the minimal polynomial coincide $\det(X\boldsymbol{I} - \boldsymbol{A}_{\mathrm{IV}}) = X^3 - X^2 - 4X + 4$, corresponding to the equation $y''' - y'' - 4y' + 4y = 0$. However, if we take the polynomial $\Psi_{\boldsymbol{A}_{\mathrm{IV}}} = X^2 - 3X + 2$ such that $\Psi_{\boldsymbol{A}_{\mathrm{IV}}}$ divides $\det(X\boldsymbol{I} - \boldsymbol{A}_{\mathrm{IV}})$ and such that—with a slight abuse of notation

  $$\Psi_{\boldsymbol{A}_{\mathrm{IV}}}(\boldsymbol{A}_{\mathrm{IV}}) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 12 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

  we see that it annihilates the first row, thus can be used instead of the other polynomials to yield (7.19). This gives $y'' - 3y' + 2y = 0$. The question is elucidated for lattice Boltzmann schemes in Section 7.6.3 and shows that asking for the annihilation of the whole matrix is too much to achieve a restatement of the equation focusing only on the first variable. This strategy is different from the previous one because not all the lines of the matrix are treated in the same way.

Let us transpose these observations to actual lattice Boltzmann schemes. A question which might arise concerns the possibility of performing better than the characteristic polynomial, in terms of number of steps in the corresponding Finite Difference scheme. There are cases, which seem quite rare according to our experience (we succeeded in finding only one special case where this happens, presented in the forthcoming pages), where the answer is positive (see Section 7.6.3 and the example therein). The conclusion that we are going to draw is the following: we cannot currently envision a systematic way of guaranteeing the minimality of the Finite Difference

scheme. This phenomenon has also been evoked by [Fučík and Straka, 2021] and our finding agrees with this work.

7.6.1 Minimal reduction in terms of time-steps

The first idea to obtain a simpler scheme is to use the minimal polynomial of $\mathbf{A}$ (or its submatrices, if needed) as done for $\boldsymbol{A}_{\mathrm{II}}$ in the previous discussion.

Definition 7.6.1: Minimal polynomial

Let $\mathcal{R}$ be a commutative ring and $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$. We define the minimal polynomial of $\mathbf{C}$, denoted $\mu_{\mathbf{C}}$ as being the monic polynomial in $\mathcal{R}[X]$ of smallest degree, thus in the form

$$\mu_{\mathbf{C}} = X^{\deg(\mu_{\mathbf{C}})} + \mathsf{q}_{\deg(\mu_{\mathbf{C}})-1} X^{\deg(\mu_{\mathbf{C}})-1} + \cdots + \mathsf{q}_1 X + \mathsf{q}_0,$$

with $(\mathsf{q}_k)_{k\in[\![0,\deg(\mu_{\mathbf{C}})]\!]} \subset \mathcal{R}$ such that

$$\mathbf{C}^{\deg(\mu_{\mathbf{C}})} + \mathsf{q}_{\deg(\mu_{\mathbf{C}})-1} \mathbf{C}^{\deg(\mu_{\mathbf{C}})-1} + \cdots + \mathsf{q}_1 \mathbf{C} + \mathsf{q}_0 \boldsymbol{I} = \mathbf{0}.$$

The characteristic and the minimal polynomial for problems set of a commutative ring are linked by a divisibility property. The proof of this result is standard.

Lemma 7.6.1: Characteristic *vs.* minimal polynomial

Let $\mathcal{R}$ be a commutative ring and $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$, then $\mu_{\mathbf{C}}$ by Definition 7.6.1 divides the characteristic polynomial $\det(X\boldsymbol{I} - \mathbf{C})$ by Definition 7.3.1. Therefore, we also have $\deg(\mu_{\mathbf{C}}) \leq \deg(\det(X\boldsymbol{I} - \mathbf{C}))$.

Unfortunately, the minimal polynomial cannot be mechanically computed by something like Algorithm 5 as for the characteristic polynomial, nor it allows to deduce some information on the Finite Difference scheme without explicitly computing it, since it does not stem from the determinant function with its peculiarities. The same reduction of Proposition 7.5.1 with $\deg(\mu_{\mathbf{A}})$ instead of q and q_k instead of c_k is possible. It can be observed that for Example 7.5.4 and Example 7.5.5, the minimal and the characteristic polynomial of the matrix $\mathbf{A}$ coincide. An example of lattice Boltzmann scheme where the minimal polynomial does not match the characteristic polynomial is provided in Chapter 9 with Section 9.4. However, we cannot present the example here because this would need to introduce an *ad hoc* way of proceeding based on the transfer of terms between the matrices $\mathbf{A}$ and $\mathbf{B}$, which is different from the general path presented before.

7.6.2 Relaxation on the equilibrium

Secondly, a more careful look at relaxation matrix $\boldsymbol{S}$ allows us to write it as

$$\boldsymbol{S} = \mathrm{diag}(\underbrace{s_1, \ldots, s_N}_{\text{conserved}}, \underbrace{s_{N+1}, \ldots, s_{N+Q}}_{\text{non conserved}}, 1, \ldots, 1),$$

where $s_i \in]0,2] \setminus \{1\}$ for $i \in [\![N+1, N+Q]\!]$ for some $Q \in \mathbb{N}$, whereas the last $q-Q-N$ relaxation parameters are equal to one, meaning that the corresponding moments exactly relax on their respective equilibrium. Without loss of generality, we have decided to put them at the end of $\boldsymbol{S}$. The fact of considering some relaxation rates equal to one is used in the so-called "regularization" models [Ladd, 1994]. We observe that the idea of setting non-conserved moments to equilibrium dates back to the age lattice gas automata, see [Sword, 1956]. In [Coreixas et al., 2019] and references therein, it is shown that "regularization" models enhance the stability features of the schemes.

In terms of matrix structure, the consequence is that the last $q-Q-N$ columns of $\mathbf{A}$ are zero, analogously to $\boldsymbol{A}_{\mathrm{III}}$ in the examples concerning ODEs. We can therefore employ the following decomposition of $\mathbf{A}$: $\mathbf{A} = \mathbf{A}_{[\![1,N+Q]\!]} + \tilde{\mathbf{A}}$, similarly to (7.29). We shall consider the characteristic polynomial of $\mathbf{A}[\![1, N+Q]\!]$ (if $N = 1$, otherwise the

characteristic polynomials of its submatrices), whereas we know that the second matrix $\tilde{\mathbf{A}}$ does not involve the last $q-Q-N$ moments (indeed, non conserved) because the corresponding columns are zero. In particular, by Corollary 7.5.1, we have that $\det(X\boldsymbol{I}_q-\mathbf{A})=X^{q-Q-N}\det(X\boldsymbol{I}_{N+Q}-\mathbf{A}[[\![1,N+Q]\!]])$. Therefore, Proposition 7.5.1 and Proposition 7.5.2 are still valid using $N+Q$ instead of q and one can use the matrix $\mathbf{A}[[\![1,N+Q]\!]]$ instead of $\mathbf{A}$ for the computation of the polynomial. The corresponding Finite Difference scheme for each conserved moment shall therefore have at most $Q+1$ steps instead of $q+1$.

Example 7.6.1. *Coming back to the setting of Example 7.5.3, assume that we consider $s_2\neq 1$ and $s_3=1$, hence having $Q=1$. Following the procedure described before, we obtain*

$$\det(X\boldsymbol{I}_2-\mathbf{A}[\{1,2\}])=X^2+\Big(-2\mathsf{S}(\mathsf{x}_1)-1+s_2\mathsf{S}(\mathsf{x}_1)+\frac{1}{3}(\mathsf{S}(\mathsf{x}_1)+2)\Big)X \\ +\Big(1+2\mathsf{S}(\mathsf{x}_1)-s_2(\mathsf{S}(\mathsf{x}_1)+1)-\frac{5}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{5}\Big)+\frac{2s_2}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{2}\Big)\Big),$$

yielding the corresponding Finite Difference scheme

$$\begin{aligned}\mathsf{m}_1(t+\Delta t)=&\Big(2\mathsf{S}(\mathsf{x}_1)+1-s_2\mathsf{S}(\mathsf{x}_1)-\frac{1}{3}(\mathsf{S}(\mathsf{x}_1)+2)\Big)\mathsf{m}_1(t)\\&+\Big(-1-2\mathsf{S}(\mathsf{x}_1)+s_2(\mathsf{S}(\mathsf{x}_1)+1)+\frac{5}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{5}\Big)-\frac{2s_2}{3}\Big(\mathsf{S}(\mathsf{x}_1)+\frac{1}{2}\Big)\Big)\mathsf{m}_1(t-\Delta t)+\frac{s_2}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2^{\mathrm{eq}}(t)\\&+\frac{1}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t)+\frac{(1-s_2)}{3\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1)\mathsf{m}_3^{\mathrm{eq}}(t-\Delta t).\end{aligned}\tag{7.32}$$

Unsurprisingly, this is (7.25) setting $s_3=1$, but is obtained by treating a smaller problem.

Observe that the fact of taking all the relaxation rates equal to one, relaxing on the equilibria, is the core mechanism of the relaxation schemes [Bouchut, 2004], where relaxation variables are merely useful for the sake of the computational scheme. In this case, there is nothing to do since the original lattice Boltzmann scheme is already in the form of a Finite Difference scheme on the conserved moments. Our way of proposing a corresponding Finite Difference scheme using characteristic polynomials is flawlessly compatible with this setting.

7.6.3 A different elimination strategy

The third idea is to proceed as for $\boldsymbol{A}_{\mathrm{IV}}$, namely looking for a polynomial which does not annihilate the whole matrix $\mathbf{A}$. To simplify the presentation, we limit ourselves to $N=1$, namely one conserved moment. We introduce this strategy to account for previous results on the subject [Ginzburg, 2009, d'Humières and Ginzburg, 2009]. Nevertheless, we shall justify its limited interest at the end of Section 7.6.3. The motivation of developing this idea has come from the following Example 7.6.2.

Example 7.6.2 (Link $\mathrm{D}_d\mathrm{Q}_{2W+1}$ two-relaxation-times schemes with magic parameters equal to 1/4)**.** *Consider the schemes introduced in [Ginzburg, 2009, d'Humières and Ginzburg, 2009], where for any spatial dimension d and considering $q=1+2W$ with $W\in\mathbb{N}^*$, which is the number of the so-called "links". The discrete velocities should be the opposite one of the other along each link, thus are such that*

$$\boldsymbol{c}_1=\mathbf{0},\qquad \boldsymbol{c}_{2r}=-\boldsymbol{c}_{2r+1},\qquad r\in[\![1,W]\!],$$

but no further constraint has to be enforced on their choice. The moment matrix $\boldsymbol{M}$ is taken to be

$$\boldsymbol{M}=\left[\begin{array}{c|cc|c|cc}1&1&1&\cdots&1&1\\\hline 0&\lambda&-\lambda&&&\\0&\lambda^2&\lambda^2&&&\\\hline\vdots&&&\ddots&&\\\hline 0&&&&\lambda&-\lambda\\0&&&&\lambda^2&\lambda^2\end{array}\right]\in\mathcal{M}_{1+2W}(\mathbb{R}),$$

where empty block blocks indicate null blocks of suitable size. Finally, one has to ensure the all the so-called "magic parameters" are equal to $1/4$ *for each link, which translates into our relaxation parameters to be*

$$s_{2r} = s, \qquad s_{2r+1} = 2 - s, \qquad r \in [\![1, W]\!],$$

with $s \in]0, 2]$ *given. Finally, one considers that one moment is conserved:* $N = 1$*. The claim in [Ginzburg, 2009] is that the corresponding Finite Difference scheme is the two-steps scheme*

$$\mathsf{m}_1(t + \Delta t) = (2 - s)\mathsf{m}_1(t) + (s - 1)\mathsf{m}_1(t - \Delta t) + \frac{s}{\lambda}\sum_{r=1}^{W} \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r}^{\mathrm{eq}}(t) + \frac{2 - s}{\lambda^2}\sum_{r=1}^{W} (\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}}) - 1)\mathsf{m}_{2r+1}^{\mathrm{eq}}(t). \tag{7.33}$$

This is claimed to be true regardless of the choice of d *and* W*. By direct inspection of the corresponding Finite Difference scheme (7.33), we can say that this reduction has been achieved using the polynomial* $\Psi_{\mathbf{A}} = X^2 + (s - 2)X + (1 - s)$*. However, it can be easily shown that* $\Psi_{\mathbf{A}}$ *does not annihilate the whole matrix* $\mathbf{A}$ *as the minimal and the characteristic polynomial do, but solely its first row. With the usual slight abuse of notation, this reads*

$$\Psi_{\mathbf{A}}(\mathbf{A}) = \begin{bmatrix} 0 & \cdots & 0 \\ \star & \cdots & \star \\ \vdots & \ddots & \vdots \\ \star & \cdots & \star \end{bmatrix}, \tag{7.34}$$

where the starred $\star$ *entries belonging to* D *are not necessarily zero.*

We define the polynomial annihilating all the first row of the matrix $\mathbf{A}$, except the very first element, which corresponds to the conserved moment m_1 that we ultimately want to keep.

Definition 7.6.2: Minimal polynomial annihilating most of the first row of A

We indicate by $\tilde{\Psi}_{\mathbf{A}} \in \mathsf{D}[X]$, which we might call "minimal polynomial annihilating most of the first row" of $\mathbf{A}$, the monic polynomial of smallest degree, thus under the form

$$\tilde{\Psi}_{\mathbf{A}} = X^{\deg(\tilde{\Psi}_{\mathbf{A}})} + \tilde{\mathsf{p}}_{\deg(\tilde{\Psi}_{\mathbf{A}})-1} X^{\deg(\tilde{\Psi}_{\mathbf{A}})-1} + \cdots + \tilde{\mathsf{p}}_1 X + \tilde{\mathsf{p}}_0,$$

such that

$$\left(\mathbf{A}^{\deg(\tilde{\Psi}_{\mathbf{A}})} + \tilde{\mathsf{p}}_{\deg(\tilde{\Psi}_{\mathbf{A}})-1}\mathbf{A}^{\deg(\tilde{\Psi}_{\mathbf{A}})-1} + \cdots + \tilde{\mathsf{p}}_1\mathbf{A} + \tilde{\mathsf{p}}_0\boldsymbol{I}\right)_{1j} = 0, \qquad j \in [\![2, q]\!].$$

By seeing the coefficients of this unknown polynomial as the unknowns of a linear system, the problem of finding $\tilde{\Psi}_{\mathbf{A}}$ can be rewritten in terms of matrices, similarly to what has to be done for computing the minimal polynomial, because it is not explicitly defined by a determinant *via* Definition 7.3.1. Consider the system of variable size

$$\begin{bmatrix} (\mathbf{A})_{12} & \cdots & (\mathbf{A}^{r-1})_{12} \\ \vdots & & \vdots \\ (\mathbf{A})_{1,Q+1} & \cdots & (\mathbf{A}^{r-1})_{1,Q+1} \end{bmatrix} \begin{bmatrix} \tilde{\mathsf{p}}_1 \\ \vdots \\ \tilde{\mathsf{p}}_{r-1} \end{bmatrix} = \begin{bmatrix} -(\mathbf{A}^{r})_{12} \\ \vdots \\ -(\mathbf{A}^{r})_{1,Q+1} \end{bmatrix}, \tag{7.35}$$

and try to find the smallest $r \in [\![1, \deg(\mu_{\mathbf{A}})]\!]$, which shall therefore be $r = \deg(\tilde{\Psi}_{\mathbf{A}})$ such that the previous system has a solution. It should be observed that the zero order coefficient $\tilde{\mathsf{p}}_0$ remains free. This under-determination comes from the fact that we do not request that $\tilde{\Psi}_{\mathbf{A}}$ annihilates the whole first row of $\mathbf{A}$. With this, we can proceed as for Proposition 7.5.1 and obtain the following result.

Proposition 7.6.1: Corresponding Finite Difference scheme for $N = 1$

Consider $N = 1$. Then the lattice Boltzmann scheme given by (7.21) corresponds to a multi-step explicit Finite Difference scheme on the conserved moment m_1 under the form

$$\mathsf{m}_1(t + \Delta t, \boldsymbol{x}) = -\sum_{k=1}^{\deg(\tilde{\Psi}_{\mathbf{A}})-1} \tilde{\mathsf{p}}_k \mathsf{m}_1(t + (1 - \deg(\tilde{\Psi}_{\mathbf{A}}) + k)\Delta t, \boldsymbol{x})$$

$$+\sum_{r=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_r(\mathbf{A}^r)_{11}\mathsf{m}_1(t+(1-\deg(\tilde\Psi_{\mathbf{A}}))\Delta t,\boldsymbol{x})+\left(\sum_{k=0}^{\deg(\tilde\Psi_{\mathbf{A}})-1}\left(\sum_{r=0}^{k}\tilde{\mathsf{p}}_{\deg(\tilde\Psi_{\mathbf{A}})+r-k}\mathbf{A}^r\right)\mathbf{Bm}^{\mathrm{eq}}(t-k\Delta t,\boldsymbol{x})\right)_1,$$

for all $t\in\Delta t[\![\deg(\tilde\Psi_{\mathbf{A}})-1,+\infty[\![$ and for all $\boldsymbol{x}\in\Delta x\mathbb{Z}^d$, where $(\tilde{\mathsf{p}}_k)_{k\in[\![0,\deg(\tilde\Psi_{\mathbf{A}})]\!]}\subset\mathsf{D}$ are the coefficients of $\tilde\Psi_{\mathbf{A}}$ from Definition 7.6.2.

Proof. By Definition 7.6.2, we have that

$$\left(\sum_{k=0}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_k\mathbf{A}^k\right)_{1,\cdot}=\left(\tilde{\mathsf{p}}_0+\sum_{r=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_r(\mathbf{A}^r)_{11},0,\dots,0\right).$$

Restarting from the proof of Proposition 7.5.1, in particular from (7.23) where c_k are changed into $\tilde{\mathsf{p}}_k$ and q is changed into $\deg(\tilde\Psi_{\mathbf{A}})$ and selecting the first row

$$\begin{aligned}\sum_{k=0}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_k\mathsf{m}_1(\tilde t+k\Delta t)&=\mathsf{m}_1(\tilde t+\deg(\tilde\Psi_{\mathbf{A}})\Delta t)+\sum_{k=1}^{\deg(\tilde\Psi_{\mathbf{A}})-1}\tilde{\mathsf{p}}_k\mathsf{m}_1(\tilde t+k\Delta t)+\tilde{\mathsf{p}}_0\mathsf{m}_1(\tilde t)\\&=\left(\sum_{k=0}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_k\mathbf{A}^k\mathbf{m}(\tilde t)\right)_1+\sum_{k=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_k\left(\sum_{r=0}^{k-1}\mathbf{A}^r\mathbf{Bm}^{\mathrm{eq}}(\tilde t+(k-1-r)\Delta t)\right)_1\\&=\left(\tilde{\mathsf{p}}_0+\sum_{r=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_r(\mathbf{A}^r)_{11}\right)\mathsf{m}_1(\tilde t)+\sum_{k=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_k\left(\sum_{r=0}^{k-1}\mathbf{A}^r\mathbf{Bm}^{\mathrm{eq}}(\tilde t+(k-1-r)\Delta t)\right)_1.\end{aligned}$$

The term $\tilde{\mathsf{p}}_0\mathsf{m}_1(\tilde t)$ simplifies on both sides of the identity, hence we are left with

$$\begin{aligned}\mathsf{m}_1(\tilde t+\deg(\tilde\Psi_{\mathbf{A}})\Delta t)=&-\sum_{k=1}^{\deg(\tilde\Psi_{\mathbf{A}})-1}\tilde{\mathsf{p}}_k\mathsf{m}_1(\tilde t+k\Delta t)+\sum_{r=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_r(\mathbf{A}^r)_{11}\mathsf{m}_1(\tilde t)\\&+\sum_{k=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_k\left(\sum_{r=0}^{k-1}\mathbf{A}^r\mathbf{Bm}^{\mathrm{eq}}(\tilde t+(k-1-r)\Delta t)\right)_1.\end{aligned}\tag{7.36}$$

The usual change of indices gives the claim. □

We see that we do not need the value of $\tilde{\mathsf{p}}_0$ to obtain the corresponding Finite Difference scheme, neither to perform the computation using $\mathbf{A}$, nor to deal with the equilibria through $\mathbf{B}$. Looking at the first two terms on the right hand side of (7.36), it is natural to consider the free value of $\tilde{\mathsf{p}}_0$ to be

$$\tilde{\mathsf{p}}_0=-\sum_{r=1}^{\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_r(\mathbf{A}^r)_{11}.$$

We also define

$$\mathsf{p}_k=\begin{cases}\tilde{\mathsf{p}}_k, & \text{if}\quad k\in[\![1,\deg(\tilde\Psi_{\mathbf{A}})]\!],\\ -\sum_{r=1}^{r=\deg(\tilde\Psi_{\mathbf{A}})}\tilde{\mathsf{p}}_r(\mathbf{A}^r)_{11}, & \text{if}\quad k=0.\end{cases}$$

This generates a polynomial, which is indeed $\tilde\Psi$ but with a precise choice of $\tilde{\mathsf{p}}_0$. We will soon give a precise characterization of this particular polynomial.

Definition 7.6.3: Minimal polynomial annihilating the first row of A

We indicate by $\Psi_{\mathbf{A}}\in\mathsf{D}[X]$, which we might call "minimal polynomial annihilating the first row" of $\mathbf{A}$, the monic polynomial of smallest degree, thus under the form

$$\Psi_{\mathbf{A}}=X^{\deg(\Psi_{\mathbf{A}})}+\mathsf{p}_{\deg(\Psi_{\mathbf{A}})-1}X^{\deg(\Psi_{\mathbf{A}})-1}+\cdots+\mathsf{p}_1X+\mathsf{p}_0,$$

such that

$$\left(\mathbf{A}^{\deg(\Psi_{\mathbf{A}})} + \mathsf{p}_{\deg(\Psi_{\mathbf{A}})-1}\mathbf{A}^{\deg(\Psi_{\mathbf{A}})-1} + \cdots + \mathsf{p}_1\mathbf{A} + \mathsf{p}_0\boldsymbol{I}\right)_{1j} = 0, \qquad j \in [\![1, q]\!]. \tag{7.37}$$

Compared to Definition 7.6.2, we are just asking the property to hold also for the very first element of the first row, namely for $j = 1$. This polynomial is $\tilde{\Psi}_{\mathbf{A}}$ for a particular choice of $\tilde{\mathsf{p}}_0$. It has been deduced from the reduction of the lattice Boltzmann scheme.

Lemma 7.6.2

The polynomial given by

$$\Psi_{\mathbf{A}} = X^{\deg(\tilde{\Psi}_{\mathbf{A}})} + \tilde{\mathsf{p}}_{\deg(\tilde{\Psi}_{\mathbf{A}})-1}X^{\deg(\tilde{\Psi}_{\mathbf{A}})-1} + \cdots + \tilde{\mathsf{p}}_1 X + \tilde{\mathsf{p}}_0 - \sum_{r=1}^{\deg(\tilde{\Psi}_{\mathbf{A}})} \tilde{\mathsf{p}}_r (\mathbf{A}^r)_{11},$$

where $(\tilde{\mathsf{p}}_k)_{k\in[\![0,\deg(\tilde{\Psi}_{\mathbf{A}})]\!]} \subset \mathsf{D}$ are the coefficients of $\tilde{\Psi}_{\mathbf{A}}$ from Definition 7.6.2 fulfills Definition 7.6.3.

Proof. We are only left to check (7.37) for $j = 1$. □

So in order to reduce the lattice Boltzmann scheme to a Finite Difference scheme using the new strategy, considering a $\tilde{\Psi}_{\mathbf{A}}$ from Definition 7.6.2 or the $\Psi_{\mathbf{A}}$ from Definition 7.6.3 is exactly the same thing. Moreover, the $\Psi_{\mathbf{A}}$ (but not a general $\tilde{\Psi}_{\mathbf{A}}$) can be linked to the minimal polynomial, essentially as the minimal polynomial is linked to the characteristic one, namely by divisibility.

Lemma 7.6.3: Minimal polynomial annihilating most of the first row of A *vs.* minimal polynomial

Let $\mu_{\mathbf{A}} \in \mathsf{D}[X]$ be the minimal polynomial of $\mathbf{A}$ according to Definition 7.6.1, then $\Psi_{\mathbf{A}}$ according to Definition 7.6.3 exists and divides the minimal polynomial $\mu_{\mathbf{A}}$. Moreover, $\deg(\Psi_{\mathbf{A}}) = \deg(\tilde{\Psi}_{\mathbf{A}}) \leq \deg(\mu_{\mathbf{A}})$.

Proof. The proof proceeds like the standard one of Lemma 7.6.1. One considers the Euclidian division between $\mu_{\mathbf{A}}$ and $\Psi_{\mathbf{A}}$: there exist $\mathsf{Q}, \mathsf{R} \in \mathsf{D}[X]$, respectively a quotient and a reminder such that

$$\mu_{\mathbf{A}} = \mathsf{Q}\Psi_{\mathbf{A}} + \mathsf{R},$$

with either $0 < \deg(\mathsf{R}) < \deg(\Psi_{\mathbf{A}})$ or $\deg(\mathsf{R}) = 0$ (constant reminder polynomial). Assume that $\mathsf{R} \not\equiv 0$, then we have, for every $j \in [\![1, q]\!]$ and with the usual slight abuse of notation

$$\underbrace{(\mu_{\mathbf{A}}(\mathbf{A}))_{1j}}_{=0} = (\mathsf{Q}(\mathbf{A}))_{1j}\underbrace{(\Psi_{\mathbf{A}}(\mathbf{A}))_{1j}}_{=0} + (\mathsf{R}(\mathbf{A}))_{1j}, \qquad \text{thus} \qquad (\mathsf{R}(\mathbf{A}))_{1j} = 0. \tag{7.38}$$

This means that R fulfills the requested property by Definition 7.6.3, contradicting the minimality of $\Psi_{\mathbf{A}}$ in terms of degree. Thus necessarily $\deg(\mathsf{R}) = 0$ so that this polynomial is constant. However, a constant polynomial cannot fulfill (7.38), except when $\mathsf{R} \equiv 0$, concluding the proof. □

Relying on the discussion that we have conducted so far, we can revisit Example 7.6.2.

Example 7.6.3 (Link $\mathrm{D}_d\mathrm{Q}_{2W+1}$ two-relaxation-times schemes with magic parameters equal to $1/4$)**.** *For the scheme introduced in Example 7.6.2, we have*

$$\mathbf{A} = \left[\begin{array}{c|cc|c|cc} 1 & \frac{1-s}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & \frac{s-1}{\lambda^2}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2})-1) & \cdots & \frac{1-s}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \frac{s-1}{\lambda^2}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}})-1) \\ \hline 0 & (1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & \frac{s-1}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\ 0 & \lambda(1-s)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & (s-1)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & (1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \frac{s-1}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) \\ 0 & & & & \lambda(1-s)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & (s-1)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}}) \end{array}\right] \in \mathcal{M}_{1+2W}(\mathsf{D}),$$

We look for the polynomial $\tilde{\Psi}_{\mathbf{A}}(X) = X^2 + \tilde{p}_1 X + \tilde{p}_0$ annihilating the first row of $\mathbf{A}$ except the very first entry, according to Definition 7.6.2. We have shown that this boils down to solving (7.35), which reads

$$\begin{bmatrix} \frac{1-s}{\lambda}A(t_{c_2}) \\ -\frac{1-s}{\lambda^2}(S(t_{c_2})-1) \\ \vdots \\ \frac{1-s}{\lambda}A(t_{c_{2W}}) \\ -\frac{1-s}{\lambda^2}(S(t_{c_{2W}})-1) \end{bmatrix} \begin{bmatrix} \tilde{p}_1 \end{bmatrix} = \begin{bmatrix} -\frac{(1-s)(2-s)}{\lambda}A(t_{c_2}) \\ \frac{(1-s)(2-s)}{\lambda^2}(S(t_{c_2})-1) \\ \vdots \\ -\frac{(1-s)(2-s)}{\lambda}A(t_{c_{2W}}) \\ \frac{(1-s)(2-s)}{\lambda^2}(S(t_{c_{2W}})-1) \end{bmatrix}.$$

The equations have the same structure for every block of two equations: thus we can find a solution by studying each block if it turns out that the solution does not depend on the block indices. Let $r \in [\![1, W]\!]$, then we solve

$$\begin{cases} \frac{1-s}{\lambda}A(t_{c_{2r}})\tilde{p}_1 = -\frac{(1-s)(2-s)}{\lambda}A(t_{c_{2r}}), \\ -\frac{1-s}{\lambda^2}(S(t_{c_{2r}})-1)\tilde{p}_1 = \frac{(1-s)(2-s)}{\lambda^2}(S(t_{c_{2r}})-1). \end{cases}$$

The solution is clearly given by $\tilde{p}_1 = (s-2)$. Since $(\mathbf{A}^2)_{11} = 1$, we obtain $p_0 = (2-s) - 1 = 1 - s$, which finally yields $\Psi(X) = X^2 + (s-2)X + (1-s)$.

This approach correctly recovers the result from [Ginzburg, 2009] following a different path. However, to our understanding, this new strategy is of moderate interest since it relies on an *ad hoc* and problem-dependent procedure illustrated by Example 7.6.3 which can be practically exploited only for highly constrained systems or for schemes of modest size (small q and/or Q). Moreover, for general schemes, it yields the same result as Proposition 7.5.1 using the characteristic polynomial (take Example 7.5.3 for instance) but passing from an inefficient approach to the computation of the polynomial instead of using the more efficient Algorithm 5. More precisely, it is advisable to utilize Algorithm 5—which cost is polynomial in the size of the matrix—instead of progressively construct the systems (7.35), try to find the minimum size with the desired property and then realizing that we found to be equal to q and thus the corresponding polynomial is the characteristic polynomial. This situation will come back to the surface in Chapter 10, when studying the issue of initialisation for lattice Boltzmann schemes.

7.7 Consistency, stability and convergence deduced from the corresponding Finite Difference scheme: the example of the D_1Q_3 scheme

Since thanks to Proposition 7.5.1 and Proposition 7.5.2, we have recast the evolution of the discrete solution pertaining to the conserved moments as multi-step Finite Difference schemes, we observe that the concepts of consistency and stability are directly inherited from those for multi-step Finite Difference schemes. This allows to construct a convergence theory like the one from the Lax theorem. Moreover, other results on Finite Difference schemes can be used.

The aim of Section 7.7 is to provide an illustration of this by considering the D_1Q_3 scheme introduced in Example 7.5.3, deriving stability conditions, convergence theorems according to the smoothness of the initial datum and provide numerical illustrations. We recall that the moment matrix $\boldsymbol{M}$ for this scheme is the one given by (1.6) and reads

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & \lambda & -\lambda \\ -2\lambda^2 & \lambda^2 & \lambda^2 \end{bmatrix}.$$

We shall consider $s_3 = 1$, hence $Q = 1$ (provided that $s_2 \neq 1$), in order to simplify the analysis by dealing with less free parameters. The target problem is the the linear advection equation at velocity $V > 0$, which reads

$$\begin{cases} \partial_t u + V\partial_x u = 0, & t \in [0, T], \quad x \in \mathbb{R}, \\ u(0, x) = u^\circ(x), & x \in \mathbb{R}. \end{cases} \tag{7.39}$$

We shall consider an acoustic scaling with $\lambda > 0$ fixed, so that m_1 approximates u and we shall take $m_2^{\mathrm{eq}}(\mathsf{m}_1) = V\mathsf{m}_1$ and $m_3^{\mathrm{eq}}(\mathsf{m}_1) = \kappa\mathsf{m}_1$ for some free parameter $\kappa \in \mathbb{R}$.

Since the problem we consider is now linear, the Fourier transform is a useful and powerful tool to analyze numerical schemes [Strikwerda, 2004, Chapter 2]. Let us introduce it. Consider $\mathcal{F} : \ell^1(\Delta x\mathbb{Z}^d) \cap \ell^2(\Delta x\mathbb{Z}^d) \to L^2([-\pi/\Delta x, \pi/\Delta x]^d)$, called "Fourier transform", defined as follows. Let $\mathsf{f} \in \ell^1(\Delta x\mathbb{Z}^d) \cap \ell^2(\Delta x\mathbb{Z}^d)$, then

$$\mathcal{F}[\mathsf{f}](\boldsymbol{\xi}) = \frac{1}{(2\pi)^{d/2}} \sum_{\boldsymbol{x} \in \Delta x\mathbb{Z}^d} \Delta x e^{-i\boldsymbol{x}\cdot\boldsymbol{\xi}} \mathsf{f}(\boldsymbol{x}), \qquad \boldsymbol{\xi} \in \left[-\frac{\pi}{\Delta x}, \frac{\pi}{\Delta x}\right]^d.$$

Sometimes, we will indicate the Fourier transform of a lattice function using a hat. We assume that the regularity assumptions shall hold for any function of which we consider the Fourier transform. The Fourier transform is extended to less regular entities by density arguments. The interest of the Fourier transform lies in the fact that it is an isometry between $\ell^2(\Delta x\mathbb{Z}^d)$ and $L^2([-\pi/\Delta x, \pi/\Delta x]^d)$, thanks to the Parseval's identity [Strikwerda, 2004, Chapter 2] and that it allows to represent the action of operators acting via the convolution product (also called filters) like the Finite Difference operators D as a multiplication on $\mathbb{C}$, see Remark 7.4.4. We can therefore represent any shift operator in the Fourier space.

Lemma 7.7.1: Shift operators in space: Fourier version

Let $\boldsymbol{z} \in \mathbb{Z}^d$ and $\mathsf{f} \in \ell^1(\Delta x\mathbb{Z}^d) \cap \ell^2(\Delta x\mathbb{Z}^d)$, then

$$\mathcal{F}[\mathsf{t}_{\boldsymbol{z}}\mathsf{f}](\boldsymbol{\xi}) = e^{-i\Delta x\boldsymbol{z}\cdot\boldsymbol{\xi}} \mathcal{F}[\mathsf{f}](\boldsymbol{\xi}), \qquad \boldsymbol{\xi} \in \left[-\frac{\pi}{\Delta x}, \frac{\pi}{\Delta x}\right]^d.$$

Therefore, the representation of the shift operator $\mathsf{t}_{\boldsymbol{z}}$ in the Fourier space is $\hat{\mathsf{t}}_{\boldsymbol{z}} = e^{-i\Delta x\boldsymbol{z}\cdot\boldsymbol{\xi}}$ and acts multiplicatively.

Proof. Let $\mathsf{f} : \Delta x\mathbb{Z}^d \to \mathbb{R}$ with $\mathsf{f} \in \ell^1(\Delta x\mathbb{Z}^d) \cap \ell^2(\Delta x\mathbb{Z}^d)$. We have, for every frequency $\boldsymbol{\xi} \in [-\pi/\Delta x, \pi/\Delta x]^d$ and using a change of variable

$$\mathcal{F}[\mathsf{t}_{\boldsymbol{z}}\mathsf{f}](\boldsymbol{\xi}) = \frac{1}{(2\pi)^{d/2}} \sum_{\boldsymbol{x} \in \Delta x\mathbb{Z}^d} \Delta x e^{-i\boldsymbol{x}\cdot\boldsymbol{\xi}} \mathsf{f}(\boldsymbol{x} - \boldsymbol{z}\Delta x) = \frac{1}{(2\pi)^{d/2}} \sum_{\boldsymbol{x} \in \Delta x\mathbb{Z}^d} \Delta x e^{-i(\boldsymbol{x}+\boldsymbol{z}\Delta x)\cdot\boldsymbol{\xi}} \mathsf{f}(\boldsymbol{x}) = e^{-i\Delta x\boldsymbol{z}\cdot\boldsymbol{\xi}} \mathcal{F}[\mathsf{f}](\boldsymbol{\xi}).$$

□

The rewrite of T and D in the Fourier space is then done in the straightforward manner, namely

$$\hat{\mathsf{T}} := \{e^{-i\Delta x\boldsymbol{z}\cdot\boldsymbol{\xi}} \ : \ \boldsymbol{z} \in \mathbb{Z}^d\}, \qquad \hat{\mathsf{D}} := \mathbb{R}\hat{\mathsf{T}},$$

where the sum and the products are the standard ones on $\mathbb{C}$. All that has been said on D holds for the new representation in the Fourier space $\hat{\mathsf{D}}$. Indeed, for any $\mathsf{d} = \sum_{\mathsf{t}\in\mathsf{T}} \alpha_{\mathsf{t}}\mathsf{t} \in \mathsf{D}$, we indicate $\hat{\mathsf{d}} := \sum_{\mathsf{t}\in\mathsf{T}} \alpha_{\mathsf{t}}\hat{\mathsf{t}} \in \hat{\mathsf{D}}$ its representative in the Fourier space. Considering a matrix $\mathbf{C} \in \mathcal{M}_q(\mathsf{D})$, its Fourier representation $\hat{\mathbf{C}} \in \mathcal{M}_q(\hat{\mathsf{D}})$ is obtained by taking the entry-wise Fourier transform of $\mathbf{C}$. Furthermore, we have that

$$\det(X\boldsymbol{I} - \mathbf{A}) = \sum_{k=0}^{q} \mathsf{c}_k X^k, \qquad \underset{\mathcal{F}^{-1}}{\overset{\mathcal{F}}{\rightleftarrows}} \qquad \det(X\boldsymbol{I} - \hat{\mathbf{A}}) = \sum_{k=0}^{q} \hat{\mathsf{c}}_k X^k,$$

where $(\mathsf{c}_k)_{k\in[\![0,q]\!]} \subset \mathsf{D}$ and $(\hat{\mathsf{c}}_k)_{k\in[\![0,q]\!]} \subset \hat{\mathsf{D}}$ are their Fourier representation.

For any multi-step scalar linear Finite Difference scheme, which we can write under the form

$$\sum_{k=0}^{q} \varphi_{q-k}\mathsf{m}_1(t + (1-k)\Delta t) = 0, \tag{7.40}$$

where $(\varphi_k)_{k\in[\![0,q]\!]} \subset \mathsf{D}$, one can introduce an amplification polynomial [Strikwerda, 2004, Chapter 4], which can

be written either in the primal space or—more often—in the Fourier space. This is

$$\Phi := \sum_{k=0}^{q} \varphi_k X^k, \qquad \hat{\Phi} := \sum_{k=0}^{q} \hat{\varphi}_k X^k. \tag{7.41}$$

We shall often write $\hat{\Phi}(\boldsymbol{\theta}, X)$, where the wave-number is $\boldsymbol{\theta} := \boldsymbol{\xi}\Delta x \in [-\pi, \pi]^d$ to emphasize the dependency of the amplification polynomial on it, through $\hat{\varphi}_k = \hat{\varphi}_k(\boldsymbol{\theta})$. Observe that since $\mathbb{C}$ is an algebraically closed field, then the amplification polynomial $\hat{\Phi}(\boldsymbol{\theta}, X)$ has q complex roots depending continuously on the wave-number $\boldsymbol{\theta}$, which we shall denote $\hat{\mathrm{g}}_k = \hat{\mathrm{g}}_k(\boldsymbol{\xi}\Delta x) = \hat{\mathrm{g}}_k(\boldsymbol{\theta})$ for $k \in [\![1, q]\!]$. These roots do not generally belong to $\hat{\mathrm{D}}$, see Remark 7.5.1. Still, they behave essentially like Finite Difference operators and are thus often called "pseudo-schemes". Like pseudo-differential operators, they are readily defined by means of the Fourier transform. For the D_1Q_3 scheme at hand, the amplification polynomial reads, see (7.32)

$$\hat{\Phi}(\theta, X) = X^2 + \left(\frac{1}{3}\left(\frac{\kappa}{\lambda^2} - 1\right) + \frac{1}{3}\left(-5 - \frac{\kappa}{\lambda^2} + 3s_2\right)\cos(\theta) + \frac{i s_2 V}{\lambda}\sin(\theta)\right)X + \frac{1 - s_2}{3}\left(2 + \frac{\kappa}{\lambda^2} + \left(1 - \frac{\kappa}{\lambda^2}\right)\cos(\theta)\right).$$

7.7.1 Consistency

Consistency and thus modified equations for the lattice Boltzmann scheme at hand can be found by applying the corresponding Finite Difference scheme (7.32) to a smooth function m_1 defined on $\mathbb{R}\times\mathbb{R}$ which equals the discrete solution m_1 at the grid points of $\Delta t\mathbb{N}\times\Delta x\mathbb{Z}^d$. This is followed by Taylor expansions. We shall see that consistency can also be studied using Fourier analysis, see [Strikwerda, 2004, Chapter 3]. We have

$$\begin{aligned}\left(1 + \frac{\Delta x}{\lambda}\partial_t + \frac{\Delta x^2}{2\lambda^2}\partial_{tt} + O(\Delta x^3)\right)m_1 &= \left(2 - s_2 + \frac{\Delta x^2}{2}\left(\frac{5}{3} - s_2\right)\partial_{xx} + O(\Delta x^4)\right)m_1 \\ &+ \left(-1 + s_2 - \frac{\Delta x^2}{6}(1 - s_2)\partial_{xx} + O(\Delta x^4)\right)\left(1 - \frac{\Delta x}{\lambda}\partial_t + \frac{\Delta x^2}{2\lambda^2}\partial_{tt} + O(\Delta x^3)\right)m_1 + \frac{s_2 V}{\lambda}(-\Delta x\partial_x + O(\Delta x^3))m_1 \\ &+ \frac{\kappa}{3\lambda^2}\left(\frac{\Delta x^2}{2}\partial_{xx} + O(\Delta x^4)\right) + \frac{(1 - s_2)\kappa}{3\lambda^2}\left(\frac{\Delta x^2}{2}\partial_{xx} + O(\Delta x^4)\right)\left(1 - \frac{\Delta x}{\lambda}\partial_t + \frac{\Delta x^2}{2\lambda^2}\partial_{tt} + O(\Delta x^3)\right)m_1.\end{aligned}$$

After some algebra and truncating at the third order

$$\begin{aligned}&m_1 + \frac{\Delta x}{\lambda}\partial_t m_1 + \frac{\Delta x^2}{2\lambda^2}\partial_{tt} m_1 + O(\Delta x^3) \\ &\qquad = m_1 + \frac{\Delta x(1 - s_2)}{\lambda}\partial_t m_1 - \frac{\Delta x s_2 V}{\lambda}\partial_x m_1 + \frac{\Delta x^2(s_2 - 1)}{2\lambda^2}\partial_{tt} m_1 + \frac{\Delta x^2(2 - s_2)}{6}\left(2 + \frac{\kappa}{\lambda^2}\right)\partial_{xx} m_1 + O(\Delta x^3),\end{aligned}$$

hence at leading order

$$\partial_t m_1 + V\partial_x m_1 = O(\Delta x).$$

Formally, we also have $\partial_{tt} m_1 = V^2\partial_{xx} m_1 + O(\Delta x)$, hence we can eliminate time derivatives higher than at first order, see [Warming and Hyett, 1974, Carpentier et al., 1997] to obtain the modified equation

$$\partial_t m_1 + V\partial_x m_1 - \lambda\Delta x\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(\frac{1}{3}\left(2 + \frac{\kappa}{\lambda^2}\right) - \frac{V^2}{\lambda^2}\right)\partial_{xx} m_1 = O(\Delta x^2). \tag{7.42}$$

In what follows, we shall consider that V and λ are fixed. One can make the numerical diffusion vanish—and thus the scheme being second-order accurate with (7.39)—if $s_2 = 2$, which is a staple of lattice Boltzmann schemes [Dubois, 2008, Graille, 2014, Junk and Rheinlander, 2008, Simonis et al., 2020], or by having $\kappa/\lambda^2 = -2 + 3V^2/\lambda^2$ for any $s_2 \in]0, 2]$.

In order to precisely quantify the regularity requirements on the initial datum u° to achieve the full order of convergence, see [Strikwerda, 2004, Chapter 10], we take advantage of the Fourier transform and of the amplification polynomial. Taking the Fourier transform of (7.39) in space, we obtain the equation

$$\partial_t \hat{u}(t, \xi) = \hat{q}(\xi)\hat{u}(t, \xi), \qquad \text{with} \qquad \hat{q}(\xi) = -iV\xi,$$

for $\xi \in \mathbb{R}$. With this, we have, see [Strikwerda, 2004, Theorem 10.6.1]:

Theorem 7.7.1: Accuracy

If a multi-step Finite Difference scheme (7.40) is accurate of order H as an approximation of (7.39), then there is a unique root $\hat{g}_1 = \hat{g}_1(\theta)$ (we index it by one for the sake of simplicity) of its amplification polynomial (7.41), defined for $|\theta| \leq \theta_0$ for some positive θ_0 such that

$$\hat{g}_1(\xi\Delta x) = 1 + \Delta t \hat{q}(\xi) + O(\Delta x^2),$$

as $\Delta x \to 0$. Moreover, there exists a non-negative integer ρ such that

$$\left| \frac{e^{\Delta t \hat{q}(\xi)} - \hat{g}_1(\xi\Delta x)}{\Delta t} \right| \leq C\Delta x^H (1 + |\xi|)^\rho.$$

In this case, the scheme is said to be accurate of order $[H, \rho]$.

Observe that in general (but exceptions might exist) $\rho = H + 1$. In particular, we shall see that ρ fixes the minimal required Sobolev regularity of the initial datum u°. The root $\hat{g}_1$ is the one concerning the consistency of the scheme (physical eigenvalue), whereas the remaining $\hat{g}_2, \ldots, \hat{g}_q$ are only numerical eigenvalues and only influence stability, without playing any physical role. For any scheme being consistent with an equation such as (7.39), with first order derivative in time, there is always one root $\hat{g}_1$ such that $\hat{g}_1(0) = 1$, see [Strikwerda, 2004, Chapter 4.2].

Proposition 7.7.1: Accuracy

The corresponding Finite Difference scheme to the lattice Boltzmann scheme in Example 7.5.3 with $s_3 = 1$ has the following accuracies according to Theorem 7.7.1.

- If $s_2 \in]0,2[$ and $\kappa/\lambda^2 \neq -2 + 3V^2/\lambda^2$, then $[H, \rho] = [1, 2]$
- If $s_2 \in]0,2[$ and $\kappa/\lambda^2 = -2 + 3V^2/\lambda^2$ or $s_2 = 2$ and any κ/λ^2, then $[H, \rho] = [2, 3]$.

Proof. For the scheme at hand, $\hat{g}_1$ is obtained as the root obtained by the formula for the solution of a quadratic equation by adding the square root of the discriminant. We have

- Let $s_2 \in]0,2[$ and $\kappa/\lambda^2 \neq -2 + 3V^2/\lambda^2$, this means that the order of accuracy is $H = 1$. We look for ρ. In the limit of $|\theta| \ll 1$, we have that

$$\hat{\Phi}(\theta, X) = X^2 + \left(-2 + \frac{iVs_2}{\lambda}\theta + \left(\frac{5}{3} + \frac{\kappa}{3\lambda^2} - s_2\right)\frac{\theta^2}{2} + O(\theta^3)\right) + X + (1 - s_2)\left(1 + \frac{1}{6}\left(-1 + \frac{\kappa}{\lambda^2}\right)\theta^2 + O(\theta^3)\right).$$

  With the help of symbolic computations, we obtain that

$$\hat{g}_1(\theta) = 1 - \frac{iV}{\lambda}\theta + \left(-\frac{1}{3}\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(2 + \frac{\kappa}{\lambda^2}\right) + \left(\frac{1}{s_2} - 1\right)\frac{V^2}{\lambda^2}\right)\theta^2 + O(\theta^3). \tag{7.43}$$

  Having $\hat{q}(\xi) = -iV\xi$, we have that in the limit of small frequencies $\exp(\Delta t \hat{q}(\xi)) = 1 - \frac{iV}{\lambda}\xi\Delta x - \frac{V^2}{\lambda^2}\frac{(\xi\Delta x)^2}{2} + O((\xi\Delta x)^3)$, hence

$$\begin{aligned}\left| \frac{e^{\Delta t \hat{q}(\xi)} - \hat{g}_1(\xi\Delta x)}{\Delta t} \right| &= \lambda \left| \left(\frac{1}{s_2} - \frac{1}{2}\right)\left(\frac{1}{3}\left(2 + \frac{\kappa}{\lambda^2}\right) - \frac{V^2}{\lambda^2}\right)\xi^2\Delta x + O(\Delta x^2) \right| \\ &\leq \lambda \left| \left(\frac{1}{s_2} - \frac{1}{2}\right)\left(\frac{1}{3}\left(2 + \frac{\kappa}{\lambda^2}\right) - \frac{V^2}{\lambda^2}\right) \right| \Delta x(1 + |\xi|^2) + O(\Delta x^2),\end{aligned}$$

  where the passage from the first to the second line is obtained by adding a positive quantity to the right hand side of the equality. From this, we deduce that the accuracy is $[H, \rho] = [1, 2]$.

- Let $s_2 = 2$ and κ be free. The expansion need to be carried one order further. Then in the limit of small wave-numbers
$$\hat{g}_1(\theta) = 1 - \frac{iV}{\lambda}\theta - \frac{V^2}{\lambda^2}\frac{\theta^2}{2} + \frac{iV}{2\lambda}\left(1 + \frac{\kappa}{\lambda^2}\right)\frac{\theta^3}{6} + O(\theta^4).$$
Using the fact that $\exp(\Delta t\hat{q}(\xi)) = 1 - \frac{iV}{\lambda}\xi\Delta x - \frac{V^2}{\lambda^2}\frac{(\xi\Delta x)^2}{2} + \frac{iV^3}{\lambda^3}\frac{(\xi\Delta x)^3}{6} + O((\xi\Delta x)^4)$, we obtain
$$\left|\frac{e^{\Delta t\hat{q}(\xi)} - \hat{g}_1(\xi\Delta x)}{\Delta t}\right| \leq \frac{|V|}{6}\left|\frac{V^2}{\lambda^2} - \frac{1}{2}\left(1 + \frac{\kappa}{\lambda^2}\right)\right|\Delta x^2(1 + |\xi|^3) + O(\Delta x^3),$$
whence the accuracy $[H, \rho] = [2, 3]$.

- Let $\kappa/\lambda^2 = -2 + 3V^2/\lambda^2$ and $s_2 \in]0, 2[$ be free. We obtain
$$\hat{g}_1(\theta) = 1 - \frac{iV}{\lambda}\theta - \frac{V^2}{\lambda^2}\frac{\theta^2}{2} + \frac{iV}{\lambda}\left(\frac{3}{s_2} - 2 - \frac{3V^2}{\lambda^2}\left(\frac{1}{s_2} - 1\right)\right)\frac{\theta^3}{6} + O(\theta^4),$$
hence
$$\left|\frac{e^{\Delta t\hat{q}(\xi)} - \hat{g}_1(\xi\Delta x)}{\Delta t}\right| \leq \frac{|V|}{6}\left|\left(\frac{3}{s_2} - 2\right)\left(1 - \frac{V^2}{\lambda^2}\right)\right|\Delta x^2(1 + |\xi|^3) + O(\Delta x^3).$$
Again, the accuracy is $[H, \rho] = [2, 3]$. Moreover, the scheme could become of accuracy $[H, \rho] = [3, 4]$ for $s_2 = 3/2$. However, the scheme would not be stable for this choice, as we shall see, thus is practically unusable.

□

7.7.2 Stability

The second ingredient to achieve the convergence of the numerical scheme—in the spirit of the Lax theorem [Strikwerda, 2004, Theorem 10.5.1]—is its stability. A scheme such as (7.40) is stable (notice that it might be conditionally stable upon the choice of time-space scaling, *etc.*) if for any final time $T > 0$, there exist a constant C_T such that
$$\|\mathsf{m}_1(t)\| \leq C_T \sum_{k=0}^{q-1} \|\mathsf{m}_1(k\Delta t)\|, \tag{7.44}$$
for a given norm, at every $t \in [\![q, n_T]\!]\Delta t$. Notice that (7.44) has to hold independently of the choice of initialization $\mathsf{m}_1(0), \ldots, \mathsf{m}_1((q-1)\Delta t)$ and of Δt, and must particularly hold when $\Delta t \to 0$, whence the number of considered steps n_T to reach T grows to infinity. As natural in the linear setting, we shall utilize the ℓ^2 norm $\|\cdot\|_{\ell^2}$ (the weighting by Δx is understood), for which [Strikwerda, 2004, Theorem 4.2.1] provides an explicit characterization, which assumptions fit the framework of the current example.

For this choice of norm, studying the roots of polynomials, in particular the amplification polynomial, is essential. We thus introduce the following Definition.

Definition 7.7.1: Schur and simple *von Neumann* polynomials

A polynomial with complex coefficients is said to be a Schur polynomial if all its roots are strictly inside the unit circle. If all the roots are inside the unit circle and those on the unit circle are simple, then the polynomial is said to be a simple *von Neumann* polynomial.

For they characterise the stability with respect to the $\|\cdot\|_{\ell^2}$ norm, we will particularly focus on simple *von Neumann* polynomials. The following iterative procedure by [Miller, 1971], which is recalled in [Strikwerda, 2004, Chapter 4] and later used—for example—by [Ginzburg, 2009, Lin et al., 2021, Barsukow and Abgrall, 2023] allows to check whether a polynomial is a simple *von Neumann* polynomial.

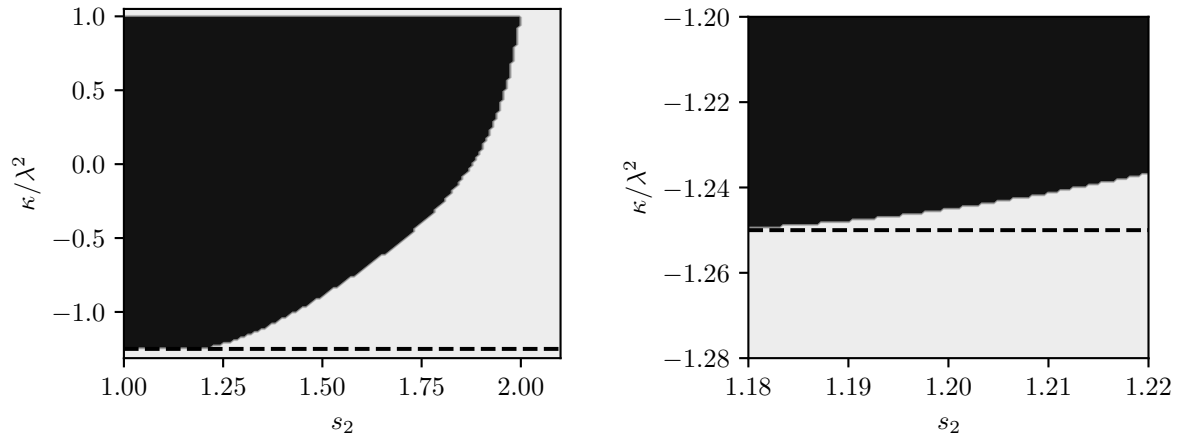


Figure 7.3: Stability region for Example 7.5.3 according to Theorem 7.7.3 with $s_3 = 1$ (in black), obtained numerically, as function of s_2 and κ/λ^2, considering $\lambda = 1$ and $V = 1/2$. The black dashed line corresponds to $\kappa/\lambda^2 = -2 + 3V^2/\lambda^2$, for which the numerical diffusivity vanishes. The right image is a magnification of the left one close to $s_2 = 1.2$.

Theorem 7.7.2: Criteria for simple *von Neumann* polynomials

A polynomial $\hat{\Phi}_r \in \mathbb{C}[X]$ of degree r is a simple von Neumann polynomial if and only if either

1. $|\hat{\Phi}_r(0)| < |\hat{\Phi}_r^\star(0)|$ and $\hat{\Phi}_{r-1}$ is a simple *von Neumann* polynomial, or
2. $\hat{\Phi}_{r-1}$ is identically zero and $\mathrm{d}_X\hat{\Phi}_r$ is Schur polynomial,

where we indicate

$$\hat{\Phi}_r^\star(X) = X^r\overline{\hat{\Phi}_r(\overline{X}^{-1})}, \qquad \text{and} \qquad \hat{\Phi}_{r-1}(X) = \frac{\hat{\Phi}_r^\star(0)\hat{\Phi}_r(X) - \hat{\Phi}_r(0)\hat{\Phi}_r^\star(X)}{X}.$$

As previously claimed, stability can be reduced to checking that the amplification polynomial is a simple *von Neumann* one. For this, we consider [Strikwerda, 2004, Theorem 4.2.1].

Theorem 7.7.3: Restricted *von Neumann* stability

If the amplification polynomial $\hat{\Phi}(\boldsymbol{\theta}, X)$ is explicitly independent of Δt and Δx, then the necessary and sufficient condition for stability of the multi-step scalar linear Finite Difference scheme (7.40), *i.e.* (7.44), for the ℓ^2 norm is that $\hat{\Phi}(\boldsymbol{\theta}, X)$ is a simple *von Neumann* polynomial for every wave-number.
Otherwise said, all the roots $\hat{\mathrm{g}}_k(\boldsymbol{\theta})$ for $k \in [\![1, q]\!]$ of $\hat{\Phi}(\boldsymbol{\theta}, X)$ given by (7.41) must satisfy the following conditions:

1. $|\hat{\mathrm{g}}_k(\boldsymbol{\theta})| \leq 1$ for every $\boldsymbol{\theta} \in [-\pi, \pi]^d$.
2. If $|\hat{\mathrm{g}}_k(\boldsymbol{\theta})| = 1$ for some $\boldsymbol{\theta} \in [-\pi, \pi]^d$, then $\hat{\mathrm{g}}_k(\boldsymbol{\theta})$ is a simple root.

Under the first condition, the Finite Difference scheme (7.40) is said to be stable in the sense of *von Neumann* with restricted condition.

We now turn to the $\mathrm{D}_1\mathrm{Q}_3$ at hand and employ the previous results. Since we shall use the method with lattice velocity $\lambda = 1$ with advection velocity $V = 1/2$, we numerically check the stability according to Theorem 7.7.3, see Figure 7.3. We observe that:

- The upper bound in the ratio $\frac{\kappa}{\lambda^2}$ is always at $\frac{\kappa}{\lambda^2} = 1$, even when changing V.
- The smaller $|V|$, the larger the stability region, as expected, because the scheme becomes more diffusive.

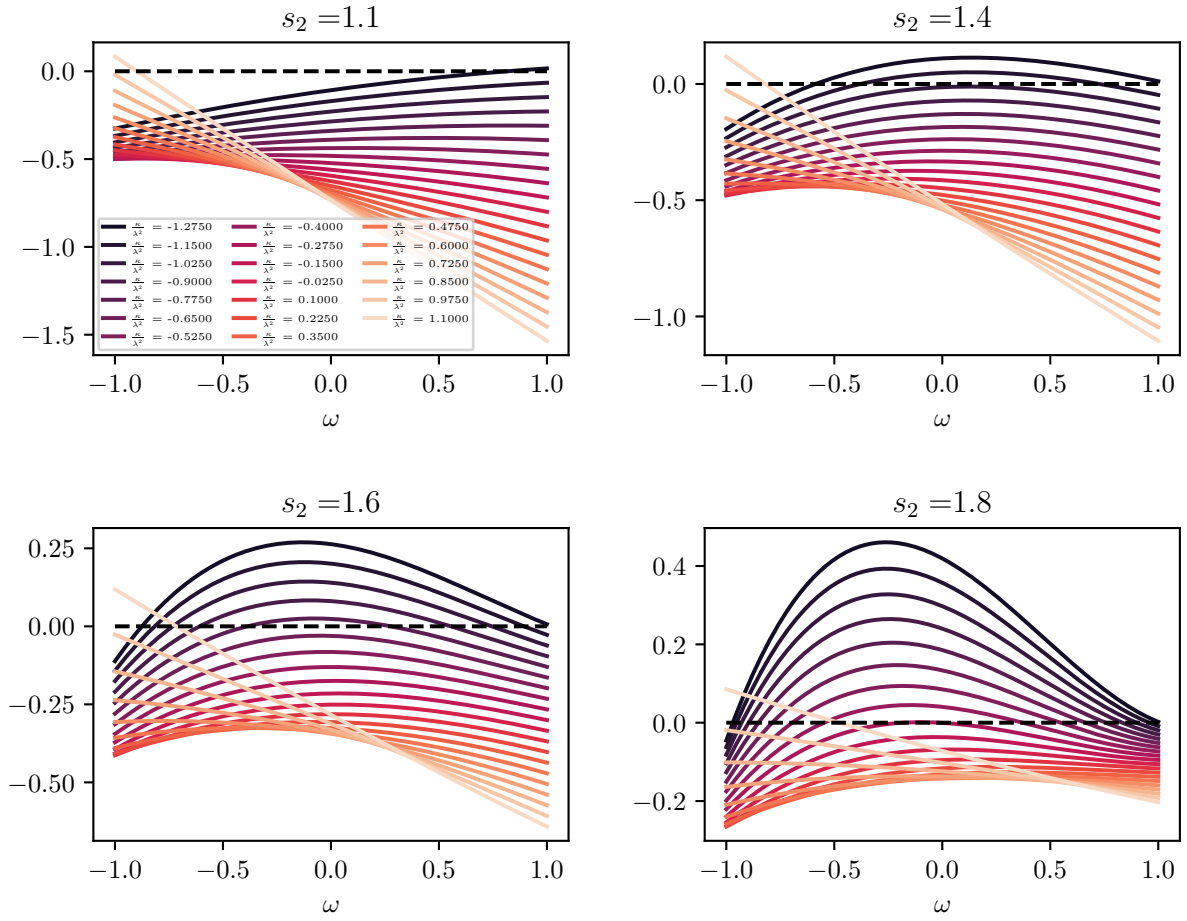


Figure 7.4: Plot of the function of $\omega = \cos(\theta)$ in inequality (7.47) for $\lambda = 1$ and $V = 1/2$ for different κ.

- Close to $s_2 = 1$, the lower bound on $\frac{\kappa}{\lambda^2}$ stays constant, whereas when increasing s_2, it increases in a continuous fashion.

We can then use Theorem 7.7.2 to find more explicit conditions, even still not completely satisfying.

Proposition 7.7.2: Necessary and sufficient conditions for stability

The corresponding Finite Difference scheme to the lattice Boltzmann scheme in Example 7.5.3 with $s_3 = 1$ is stable in the ℓ^2 sense according to Theorem 7.7.3 if and only if

$$\text{when} \quad -2 + \frac{3V^2}{\lambda^2} \leq \frac{\kappa}{\lambda^2} \leq 1, \quad \text{also} \quad -\frac{1}{2}\left(\frac{3}{|1-s_2|}+1\right) \leq \frac{\kappa}{\lambda^2} \leq \frac{1}{2}\left(\frac{3}{|1-s_2|}-1\right) \tag{7.45}$$

$$\text{when} \quad \frac{\kappa}{\lambda^2} > 1, \quad \text{also} \quad 0 < s_2 \leq 2, \quad \text{and} \tag{7.46}$$

$$\max_{\omega\in[-1,1]} \frac{s_2^2 V^2}{\lambda^2}(1+\omega)(1+\Omega(\omega))^2 + \frac{1}{9}(2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\Omega(\omega))^2\left((2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\omega) - 6(1+\Omega(\omega))\right) \leq 0, \tag{7.47}$$

with $\Omega(\cos(\theta)) := (1-s_2)(2+\kappa/\lambda^2 + (1-\kappa/\lambda^2)\cos(\theta))/3$.

Observe that in principle all the constraints (7.45), (7.46) and (7.47) are not necessarily independent. Indeed, we see that plotting the function of $\cos(\theta)$ minimized in (7.47), when we violate the upper bound in the first

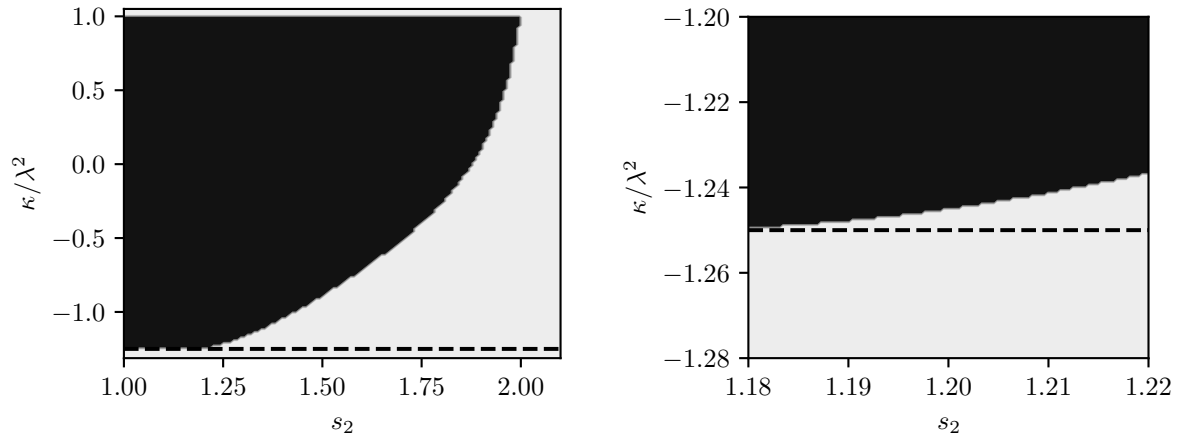


Figure 7.5: Validity region of the inequality (7.47), obtained numerically, as function of s_2 and κ/λ^2, considering $\lambda = 1$ and $V = 1/2$. The black dashed line corresponds to $\kappa/\lambda^2 = -2 + 3V^2/\lambda^2$, for which the numerical diffusivity vanishes. The right image is a magnification of the left one close to $s_2 = 1.2$.

constraint out of (7.45), the maximum (of a linear decreasing function) violates the inequality and is located at $\cos(\theta) = -1$ (high frequencies), which shows that as previously remarked, (7.45) and (7.47) are not independent. Then, decreasing κ/λ^2, a local maximum starts to form inside $[-1,1]$ and typically yields stability. However, the behavior is different when s_2 is close to one and when it is way larger than one. For s_2 close to one (*i.e.* $s_2 = 1.1$), once decreasing κ/λ^2, a local maximum forms inside $[-1,1]$. Still, the function in (7.47) always remain negative, yielding stability. At some moment, the maximum is reached for $\cos(\theta) = 1$ (low frequency, thus the fact that this constraint corresponds to having positive numerical dissipation) and and decreasing κ/λ^2 once more yields instability, because (indeed) it violates the lower bound in (7.45). Again, the constraints show not to be independent. Quite the opposite, for s_2 away from one (*i.e.* $s_2 = 1.4, 1.6$ and 1.8), once the local maximum is inside $[-1,1]$, it never exits from this compact set when decreasing κ/λ^2, until it violates the inequality. This explains the different behavior of the lower bound for κ/λ^2 in Figure 7.3 according to the choice of s_2. The maximum can be reached either on the boundary of $[-1,1]$ (in particular at 1, *i.e.* low frequencies) (for $s_2 \leq 1.18$ approximately) yielding the flat profile close to $s_2 = 1$, or inside this compact set (for $s_2 > 1.18$), giving the tightening shape as s_2 increases towards $s_2 = 2$, due to frequencies in between low and high. Overall, the constraint (7.47) seems to encompass all the remaining ones, namely (7.45) and (7.46), which is empirically confirmed by Figure 7.5, compared to Figure 7.3: the whole stability domain is the one represented by (7.47).

Proof of Proposition 7.7.2. First, observe that coherently with [Warming and Hyett, 1974]—even if our setting cannot allow to deduce (quite the opposite, this is false) that having positive dissipation in the small wave-number limit is necessary and sufficient to achieve stability—a necessary condition is the positivity of numerical dissipation. This comes from the low wave-number expansion (7.43) of the consistency eigenvalue, which leads in the limit of $|\theta| \ll 1$

$$|\hat{\mathrm{g}}_1(\theta)|^2 = 1 - 2\left(\frac{1}{s_2} - \frac{1}{2}\right)\left(\frac{1}{3}\left(2 + \frac{\kappa}{\lambda^2}\right) - \frac{V^2}{\lambda^2}\right)\theta^2 + O(\theta^3),$$

which imposes

$$\frac{\kappa}{\lambda^2} \geq -2 + \frac{3V^2}{\lambda^2}, \tag{7.48}$$

because otherwise the eigenvalue $\hat{\mathrm{g}}_1(\theta)$ would be initial increasing in modulus for small wave-numbers, thus being of modulus strictly larger than one in a neighborhood of $\theta = 0$. In order to find other conditions for stability, we start from the amplification polynomial $\hat{\Phi}$, in an iterative fashion as prescribed by Theorem 7.7.2:

$$\hat{\Phi}_2(\theta, X) = X^2 + \left(\frac{1}{3}\left(\frac{\kappa}{\lambda^2} - 1\right) + \frac{1}{3}\left(-5 - \frac{\kappa}{\lambda^2} + 3s_2\right)\cos(\theta) + \frac{i s_2 V}{\lambda}\sin(\theta)\right)X + \frac{1 - s_2}{3}\left(2 + \frac{\kappa}{\lambda^2} + \left(1 - \frac{\kappa}{\lambda^2}\right)\cos(\theta)\right).$$

We introduce

$$\hat{\Phi}_2^\star(\theta,X) = X^2\overline{\hat{\Phi}_2(\theta,\overline{X}^{-1})}$$
$$= \frac{1-s_2}{3}\left(2+\frac{\kappa}{\lambda^2}+\left(1-\frac{\kappa}{\lambda^2}\right)\cos(\theta)\right)X^2+\left(\frac{1}{3}\left(\frac{\kappa}{\lambda^2}-1\right)+\frac{1}{3}\left(-5-\frac{\kappa}{\lambda^2}+3s_2\right)\cos(\theta)-\frac{is_2V}{\lambda}\sin(\theta)\right)X+1.$$

We obtain

$$\hat{\Phi}_2(\theta,0) = \frac{1-s_2}{3}\left(2+\frac{\kappa}{\lambda^2}+\left(1-\frac{\kappa}{\lambda^2}\right)\cos(\theta)\right), \qquad \hat{\Phi}_2^\star(\theta,0)=1.$$

The first condition to check is

$$|\hat{\Phi}_2(\theta,0)|^2-|\hat{\Phi}_2^\star(\theta,0)|^2 = \frac{(1-s_2)^2}{9}\left(2+\frac{\kappa}{\lambda^2}+\left(1-\frac{\kappa}{\lambda^2}\right)\cos(\theta)\right)^2-1<0, \tag{7.49}$$

for every $\theta\in[-\pi,\pi]$. (7.49) is a quadratic inequality in the unknown $\cos(\theta)\in[-1,1]$ with positive coefficient for the second-order term. This means that the maximum of the left hand side is reached on the boundary of $[-1,1]$. Let us determine which point on the boundary is maximal: we study

$$\left[\frac{(1-s_2)^2}{9}\left(2+\frac{\kappa}{\lambda^2}+\left(1-\frac{\kappa}{\lambda^2}\right)\cos(\theta)\right)^2-1\right]_{\theta=0} = (1-s_2)^2-1$$
$$\leq \left[\frac{(1-s_2)^2}{9}\left(2+\frac{\kappa}{\lambda^2}+\left(1-\frac{\kappa}{\lambda^2}\right)\cos(\theta)\right)^2-1\right]_{\theta=\pm\pi} = \frac{(1-s_2)^2}{9}\left(1+\frac{\kappa}{\lambda^2}\right)-1,$$

which becomes, for $s_2\neq 1$

$$-2\leq\frac{\kappa}{\lambda^2}\leq 1, \qquad \text{thus with the previous constraint} \qquad -2+\frac{3V^2}{\lambda^2}\leq\frac{\kappa}{\lambda^2}\leq 1,$$

hence we study the maximum for $\cos(\theta)=-1$, yielding

$$-\frac{1}{2}\left(\frac{3}{|1-s_2|}+1\right)\leq\frac{\kappa}{\lambda^2}\leq\frac{1}{2}\left(\frac{3}{|1-s_2|}-1\right).$$

Otherwise, for

$$\frac{\kappa}{\lambda^2}<-2 \quad \text{or} \quad \frac{\kappa}{\lambda^2}>1, \qquad \text{thus with the previous constraint} \qquad \frac{\kappa}{\lambda^2}>1,$$

we study the maximum for $\cos(\theta)=1$, giving $s_2\in]0,2]$. We are ready to compute $\hat{\Phi}_1$ which is given, after long computations, by

$$\hat{\Phi}_1(\theta,X) = (1-\Omega(\cos(\theta))^2)X-(1-\Omega(\cos(\theta)^2)+\frac{1}{3}(2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\cos(\theta))(1-\Omega(\cos(\theta)))$$
$$+\frac{is_2V}{\lambda}\sin(\theta)(1+\Omega(\cos(\theta))),$$

where we have defined

$$\Omega(\cos(\theta)) := \frac{1-s_2}{3}\left(2+\frac{\kappa}{\lambda^2}+\left(1-\frac{\kappa}{\lambda^2}\right)\cos(\theta)\right).$$

Requesting that $\hat{\Phi}_1$ is simple von Neumann polynomial reads, by computing the square of the modulus of its unique root and using the fact that $\sin^2(\theta)=1-\cos^2(\theta)=(1+\cos(\theta))(1-\cos(\theta))$.

$$\frac{s_2^2V^2}{\lambda^2}(1-\cos(\theta))(1+\cos(\theta))(1+\Omega(\cos(\theta)))^2+\frac{1}{9}(2-s_2)^2\left(2+\frac{\kappa}{\lambda^2}\right)^2(1-\cos(\theta))^2(1-\Omega(\cos(\theta)))^2$$
$$-\frac{2}{3}(2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\cos(\theta))(1-\Omega(\cos(\theta)))(1-\Omega(\cos(\theta))^2)\leq 0,$$

which gives, after simplifications

$$\frac{s_2^2V^2}{\lambda^2}(1+\cos(\theta))(1+\Omega(\cos(\theta)))^2+\frac{1}{9}(2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\Omega(\cos(\theta)))^2\left((2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\cos(\theta))\right.$$
$$\left.-6(1+\Omega(\cos(\theta)))\right)\leq 0.$$

This is a third-order polynomial inequality in $\cos(\theta)\in[-1,1]$. Unfortunately, the left hand side sometimes reaches its maximum value on the boundary of $[-1,1]$ and sometimes inside, making it difficult to provide more explicit conditions, apart from

$$\max_{\omega\in[-1,1]}\frac{s_2^2V^2}{\lambda^2}(1+\omega)(1+\Omega(\omega))^2+\frac{1}{9}(2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\Omega(\omega))^2\left((2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\omega)-6(1+\Omega(\omega))\right)\leq 0.$$

□

We can also write more handy necessary conditions, which empirically turn out to be also necessary in the cases that we analyzed.

Proposition 7.7.3: Sufficient conditions for stability

The corresponding Finite Difference scheme to the lattice Boltzmann scheme in Example 7.5.3 with $s_3=1$ is stable in the ℓ^2 sense according to Theorem 7.7.3 if

$$\frac{|V|}{\lambda}\leq 1,\qquad 0<s_2\leq 2,\qquad -2+\frac{3V^2}{\lambda^2}\leq\frac{\kappa}{\lambda^2}\leq 1,\qquad \text{and}$$
$$\max_{\omega\in[-1,1]}\frac{s_2^2V^2}{\lambda^2}(1+\omega)(1+\Omega(\omega))^2+\frac{1}{9}(2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\Omega(\omega))^2\left((2-s_2)\left(2+\frac{\kappa}{\lambda^2}\right)(1-\omega)\right.$$
$$\left.-6(1+\Omega(\omega))\right)\leq 0,\quad (7.50)$$

with $\Omega(\cos(\theta)):=(1-s_2)(2+\kappa/\lambda^2+(1-\kappa/\lambda^2)\cos(\theta))/3$.

The first condition in (7.50) is the standard CFL condition. The second one is the standard bound on the relaxation parameters for a lattice Boltzmann scheme and finally the third condition selects an interval for κ/λ^2 between positive dissipation (lower bound) and an upper bound pertaining to high frequencies.

Proof of Proposition 7.7.3. (7.48) is found as in the proof of Proposition 7.7.2. Then, the quadratic inequality (7.49) is studied at the extremal values for $\cos(\theta)=\pm1$ regardless of which one is the actual maximum, hence the sufficient character.

- Considering (7.49) for $\theta=0$, hence for $\cos(\theta)=1$, corresponding to a low frequency stability, we obtain $(1-s_2)^2-1<0$, which gives
$$s_2\in]0,2[.$$
- Considering (7.49) for $\theta=\pm\pi$, hence for $\cos(\theta)=-1$, corresponding to a high frequency stability, we gain
$$\frac{(1-s_2)^2}{9}\left(1+\frac{2\kappa}{\lambda^2}\right)^2-1<0,\qquad \text{hence}\qquad -\frac{1}{2}\left(\frac{3}{|1-s_2|}+1\right)<\frac{\kappa}{\lambda^2}<\frac{1}{2}\left(\frac{3}{|1-s_2|}-1\right).$$
Of course we assume $s_2\neq1$, otherwise everything is trivial. For the lower bound on κ/λ^2, we observe that $3/|1-s_2|\geq3$, hence the bound is necessarily satisfied under the condition $\kappa/\lambda^2>-2$. Still, this condition is redundant since we have assumed that (7.48) holds. Going to the upper bound, again because $3/|1-s_2|\geq3$, we can fulfill it by checking that
$$\frac{\kappa}{\lambda^2}<1,\quad (7.51)$$
which is a non-trivial condition. Observe that in order to have some room for the ration κ/λ^2 according to (7.48) and (7.51), the CFL condition $|V|/\lambda\leq1$ must hold.

The last part of the proof of Proposition 7.7.2 remains the same. □

7.7.3 Convergence and numerical experiments

Using well-known results on multi-step Finite Difference scheme, one obtains the following convergence result for the original lattice Boltzmann scheme that we consider in Section 7.7. The aim is not to state and prove the most general result, yet the one which can be easily deduced from well-established ones without additional effort, see [Strikwerda, 2004].

Proposition 7.7.4: Convergence

Consider the lattice Boltzmann scheme in Example 7.5.3 with $s_3 = 1$ for a choice of $(\lambda, V, \kappa, s_2)$ rendering a ℓ^2 stable scheme, as discussed in Proposition 7.7.2 and Proposition 7.7.3. The scheme is initialized with the point values of u° and at equilibrium. Then

- For $s_2 \in]0,2[$ and $\kappa/\lambda^2 \neq -2 + 3V^2/\lambda^2$, namely the corresponding Finite Difference is accurate at order $[H, \rho] = [1,2]$.
  - If $u^\circ \in H^2(\mathbb{R}) = H^\rho(\mathbb{R})$, the convergence of the lattice Boltzmann scheme towards the solution of (7.39) is linear:
  $$\|u(t, \Delta x \mathbb{Z}) - \mathsf{m}_1(t)\|_{\ell^2} \leq C\Delta x \|u^\circ\|_{H^2(\mathbb{R})}, \qquad t \in [\![0, n_T]\!]\Delta t,$$
  where $u(t, \Delta x\mathbb{Z})$ stands for the exact solution of (7.39) evaluated at the lattice points.
  - If $u^\circ \in H^\sigma(\mathbb{R})$ for every $\sigma < \sigma_0 < 2 = \rho$ and there exists a constant $\tilde{C}(u^\circ)$ such that $\|u^\circ\|_{H^\sigma(\mathbb{R})} \leq \tilde{C}(u^\circ)/\sqrt{\sigma_0 - \sigma}$, and whenever $\sigma_0 < 1+$, assume that the initial datum u° is a piece-wise differentiable function except at a finite number of jump discontinuities $x_1, \ldots, x_r$ such that
  $$\int_{|x|>K} \left(|u^\circ(x)|^2 + |\mathrm{d}_x u^\circ(x)|^2\right)\mathrm{d}x < +\infty, \tag{7.52}$$
  for $K > \max(|x_1|, |x_r|)$. The convergence of the lattice Boltzmann scheme towards the solution of (7.39) is done with a reduced rate:
  $$\|u(t, \Delta x \mathbb{Z}) - \mathsf{m}_1(t)\|_{\ell^2} \leq C\Delta x^{\sigma_0/2}\sqrt{|\ln(\Delta x)|}\tilde{C}(u^\circ), \qquad t \in [\![0, n_T]\!]\Delta t.$$
- For $s_2 \in]0,2[$ and $\kappa/\lambda^2 = -2 + 3V^2/\lambda^2$ or $s_2 = 2$ and any κ/λ^2, namely the corresponding Finite Difference is accurate at order $[H, \rho] = [2,3]$.
  - If $u^\circ \in H^3(\mathbb{R}) = H^\rho(\mathbb{R})$, the convergence of the lattice Boltzmann scheme towards the solution of (7.39) is quadratic:
  $$\|u(t, \Delta x \mathbb{Z}) - \mathsf{m}_1(t)\|_{\ell^2} \leq C\Delta x^2 \|u^\circ\|_{H^3(\mathbb{R})}, \qquad t \in [\![0, n_T]\!]\Delta t.$$
  - If $u^\circ \in H^\sigma(\mathbb{R})$ for every $\sigma < \sigma_0 < 3 = \rho$ and there exists a constant $\tilde{C}(u^\circ)$ such that $\|u^\circ\|_{H^\sigma(\mathbb{R})} \leq \tilde{C}(u^\circ)/\sqrt{\sigma_0 - \sigma}$, and whenever $\sigma_0 < 1+$, assume that the initial datum u° is a piece-wise differentiable function except at a finite number of jump discontinuities $x_1, \ldots, x_r$ satisfying (7.52), the convergence of the lattice Boltzmann scheme towards the solution of (7.39) is done with a reduced rate:
  $$\|u(t, \Delta x \mathbb{Z}) - \mathsf{m}_1(t)\|_{\ell^2} \leq C\Delta x^{2\sigma_0/3}\sqrt{|\ln(\Delta x)|}\tilde{C}(u^\circ), \qquad t \in [\![0, n_T]\!]\Delta t.$$

The constants C have the following dependencies: $C = C(T, \lambda, V, \kappa, s_2)$.

Remark 7.7.1 (Assumptions in Proposition 7.7.4). *The assumption concerning the existence of the constant $\tilde{C}(u^\circ)$ comes by the fact that we want to use the result for $u^\circ \in H^{\sigma_0-}(\mathbb{R})$ and not in $H^{\sigma_0}(\mathbb{R})$. However, the price to pay is the factor $\sqrt{|\ln(\Delta x)|}$ in the estimation. Observe that—as pointed out in [Strikwerda, 2004, Chapter 10.3]—the factor*

$\sqrt{|\ln(\Delta x)|}$ is rarely observed throughout numerical simulations. The technical assumption on the jump discontinuities and (7.52) means that u° is in H^1 outside the compact set containing the discontinuities. It is used in order to define the pointwise evaluation of the initial datum and the exact solution when they are less than $H^1(\mathbb{R}) \subset C^0(\mathbb{R})$ (see [Brézis, 2011, Theorem 9.12 and Remark 11]). Observe that analogous convergence rates can be deduced following the techniques by [Courtès, 2017, Chapter 2], using initial data being the averages of u° and comparing to the averages of the exact solution. The same convergence rates can also be obtained using estimates within Besov spaces, see [Brenner et al., 1975].

Proof of Proposition 7.7.4. This proof relies on well-known results which are reused. Since, according to Proposition 7.7.1, the corresponding multi-step Finite Difference scheme can be up to $[H, \rho] = [2, 3]$ accurate and $\Delta t \propto \Delta x$, according to [Strikwerda, 2004, Theorem 10.6.2], it must be initialized with an initialization scheme of order at least $H = 1$ in order not to lower its order. This is achieved, as we shall explain in Chapter 10, by taking the initial datum at equilibrium.

In the regular case where $u^\circ \in H^\rho(\mathbb{R})$, we simply apply the generalization of [Strikwerda, 2004, Theorem 10.1.4] to multi-step schemes, since the assumption that $\rho > 1/2$ and $H \leq \rho$ is always fulfilled for our scheme, see Proposition 7.7.1. For non-smooth initial data, we just apply [Strikwerda, 2004, Corollary 10.3.2], with the assumption on the jump discontinuities and the technical assumption (7.52) in order to apply [Strikwerda, 2004, Corollary 10.3.3]. □

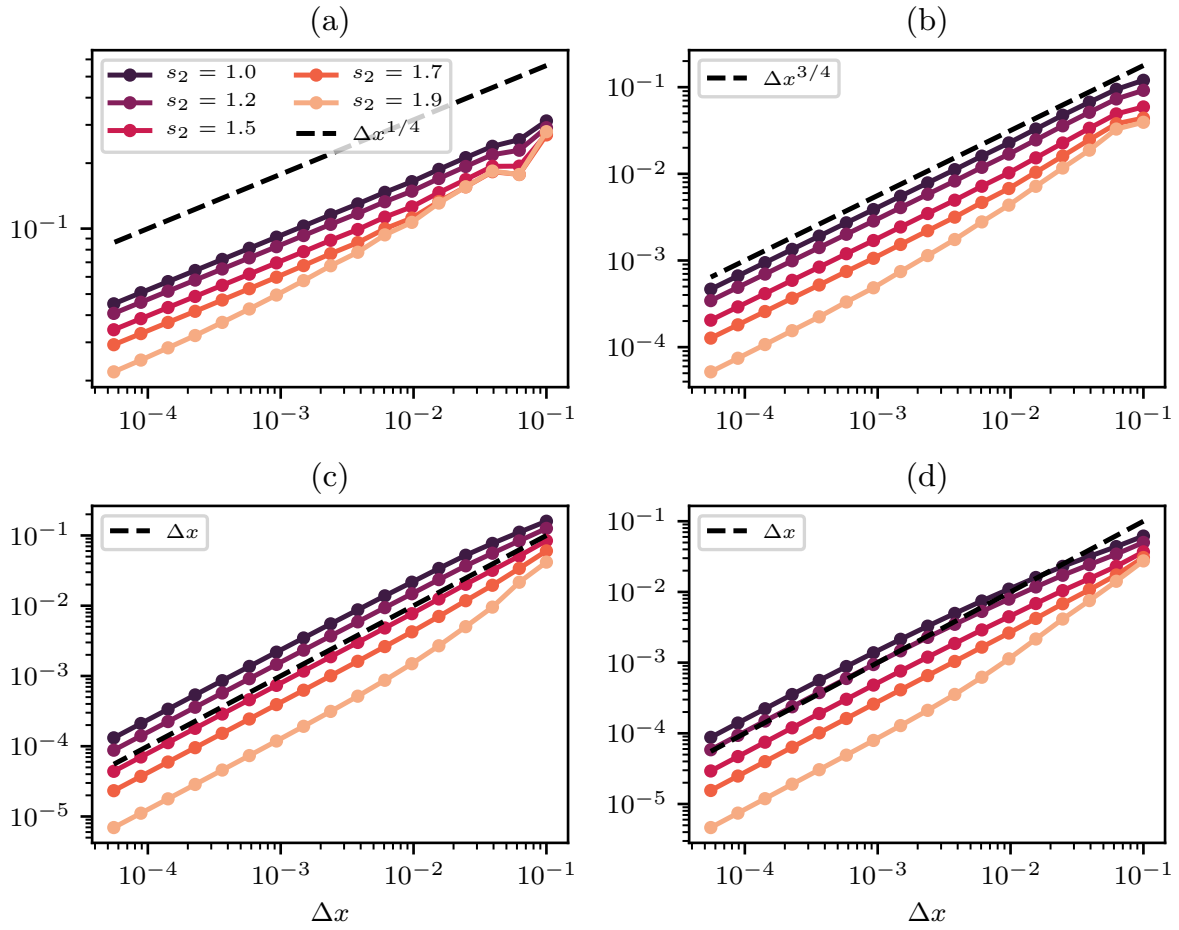


Figure 7.6: $\kappa/\lambda^2 = 0.8$. Error $\|u(T, \Delta x\,\mathbb{Z}) - \mathsf{m}_1(T)\|_{\ell^2}$ at final time T: error between the solution (conserved moment) of lattice Boltzmann scheme and the exact solution, for different initial data (a), (b), (c) and (d) and different relaxation parameters s_2.

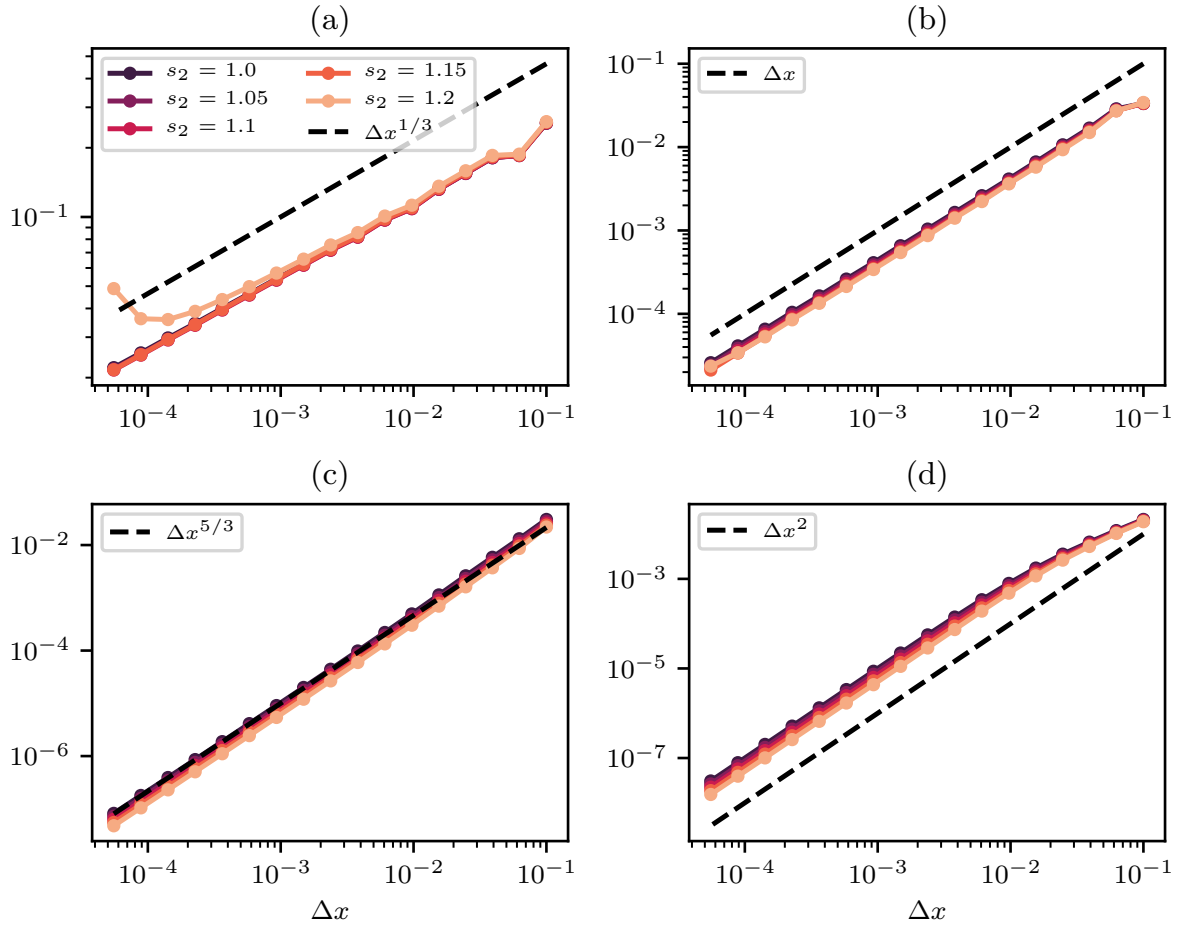


Figure 7.7: $\kappa/\lambda^2 = -1.25$. Error $\|u(T, \Delta x\mathbb{Z}) - \mathsf{m}_1(T)\|_{\ell^2}$ at final time T: error between the solution (conserved moment) of lattice Boltzmann scheme and the exact solution, for different initial data (a), (b), (c) and (d) and different relaxation parameters s_2.

We now corroborate Proposition 7.7.4 with numerical simulations, which are carried, for the sake of the numerical implementation, on the bounded domain $\Omega = [-1, 1]$ enforcing periodic boundary conditions. The final simulation time is $T = 1/2$ and $\lambda = 1$, fixing $V = 1/2$. We stress the fact that we employ the lattice Boltzmann scheme and not its corresponding Finite Difference scheme. The conserved moment is initialized using the point values of the initial condition. The non-conserved data are initialized at equilibrium. Guided by the considerations from Proposition 7.7.4 in terms of regularity, we take different initial functions with various smoothness, inspired by [Strikwerda, 2004, Courtès, 2017].

$$\text{(a)} \quad u^\circ(x) = \mathbb{1}_{[0,1/2]}(|x|) \in H^\sigma(\mathbb{R}), \qquad \text{for any} \quad \sigma < \sigma_0 = 1/2. \tag{7.53}$$

$$\text{(b)} \quad u^\circ(x) = (1 - 2|x|)\mathbb{1}_{[0,1/2]}(|x|) \in H^\sigma(\mathbb{R}), \qquad \text{for any} \quad \sigma < \sigma_0 = 3/2. \tag{7.54}$$

$$\text{(c)} \quad u^\circ(x) = \cos^2(\pi x)\mathbb{1}_{[0,1/2]}(|x|) \in H^\sigma(\mathbb{R}), \qquad \text{for any} \quad \sigma < \sigma_0 = 5/2. \tag{7.55}$$

$$\text{(d)} \quad u^\circ(x) = \exp(-1/(1 - |2x|^2))\mathbb{1}_{[0,1/2]}(|x|) \in C_c^\infty(\mathbb{R}). \tag{7.56}$$

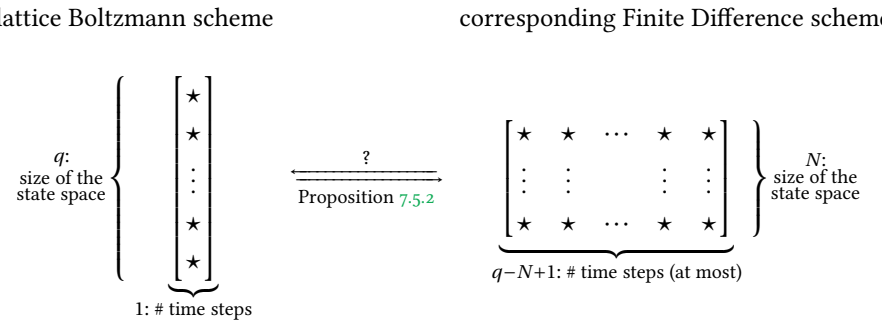
This data obey the assumptions of Proposition 7.7.4. The numerical convergence for the case $\kappa/\lambda^2 = 0.8$ is given on Figure 7.6. According to Figure 7.3 and Proposition 7.7.2, we expect stability for every choice of s_2. Thus, the empirical convergence rates are in excellent agreement with Proposition 7.7.4. The error constant is smaller for larger s_2, since for this choice, less numerical diffusion is present.

Concerning the case $\kappa/\lambda^2 = -1.25$ presented on Figure 7.7, we had to utilize relaxation parameters s_2 close to one in order to remain in the stability region as prescribed by Figure 7.3 and Proposition 7.7.2. As far as the scheme

stays stable, for $s_2 \leq 1.15$, we observe the expected convergence rates according to Proposition 7.7.4. Nevertheless, looking at the right image in Figure 7.3, we see that $s_2 = 1.2$ is not in the stability region. This is why we observe, in (a) from Figure 7.3, thus for the less smooth solution, that the scheme is not convergent. The instability originates from high-frequency modes which are abundant in the test case (a). This is the empirical evidence that the Lax-Richtmyer theorem [Lax and Richtmyer, 1956] holds for lattice Boltzmann schemes: an unstable scheme cannot be convergent.

7.8 Conclusions of Chapter 7

In this Chapter 7, we have eliminated the non-conserved moments from any lattice Boltzmann scheme at hand, at a fully discrete level. This was dictated by the fact that these quantities do not have a continuous analogue in the problem to solve and thus complicate the task of defining consistency and stability, thus deducing convergence. To this end, we have introduced a suitable commutative ring to represent lattice Boltzmann schemes, which allows to apply the Cayley-Hamilton theorem and recast the scheme, as far as the discrete dynamics of the conserved moments is concerned, as multi-step Finite Difference schemes solely on these variables. However, it is important to emphasize that viewing lattice Boltzmann schemes as Finite Difference scheme must be seen as a tool for theoretical analysis and not as the right way of implementing them. For example, the original formulation of the lattice Boltzmann schemes is highly suitable for parallelization and the implementation of the stream phase can be strongly optimized. On the other hand, its Finite Difference counterpart cannot easily handle these optimizations, showing that the original formulation of the lattice Boltzmann schemes is the right choice when implementation is concerned. Indeed, the original lattice Boltzmann can be seen as a strongly optimized implementation of the corresponding Finite Difference scheme introduced in Chapter 7. The result from Proposition 7.5.2 acts as a one-way mathematical transform in the following manner:



Observe that the result from Proposition 7.5.2 is essentially a way of dealing with time in a different way: the number of variables is reduced at the price of adding time steps. Otherwise said, the role of Proposition 7.5.2 is to "flip" the previous matrices, "exchanging" non-conserved moments relaxing away from their equilibrium for time steps.

Therefore, the lattice Boltzmann schemes inherit the notions of consistency with respect to a given problem and stability from the theory of multi-step Finite Difference schemes. These two notions are crucial to deduce the convergence of the schemes. In particular, one can easily deduce—in a linear setting—the Lax-Richtmyer theorem [Lax and Richtmyer, 1956, Strikwerda, 2004]. We showcased that for a linear D_1Q_3, considering the ℓ^2 norm, we can study the consistency of the scheme and provide stability conditions as well as precise convergence orders according to the regularity of the initial datum, just by reusing known results on Finite Difference schemes.

However, this was done by explicitly writing down the corresponding Finite Difference scheme for the D_1Q_3 at hand and might become cumbersome for more complicated schemes, as predicted in [Junk and Yang, 2015]. Then, the question we are left to answer is whether the one-way transform by Proposition 7.5.2 is sufficiently well-characterized in order to perform the consistency—*cf.* Chapter 8—and stability—*cf.* Chapter 9—analyses on the original lattice Boltzmann scheme without explicitly computing the corresponding Finite Difference scheme, yet benefitting from the rigorous setting of Finite Difference schemes.

Chapter 8

Consistency and modified equations

General context and motivation

Consistency is the first cornerstone to deduce the convergence of a numerical scheme. Indeed, the numerical solution of a scheme which is non-consistent with the equations at hand shall hardly converge to the solution of the latter. Still, numerical methods are intrinsically different from the equations they aim at solving, because: "Finite difference approximations have a more complicated "physics" than the equations they are designed to simulate. The irony is no paradox, however, for finite differences are used not because the numbers they generate have simple properties, but because those numbers are simple to compute", see [Trefethen, 1996, Chapter 5]. The method of the modified equations [Warming and Hyett, 1974] and [Gustafsson et al., 1995, Strikwerda, 2004, Carpentier et al., 1997] has proved to be a valuable tool to describe such "complicated physics". Concerning lattice Boltzmann schemes, consistency is not an obvious matter and the question boils down to studying towards which solution they converge. We have observed that this can be done by relying on the corresponding Finite Difference schemes. However, we would like to be able to do this without computing them explicitly for the lattice Boltzmann method at hand.

State of the art

The standpoint of the lattice Boltzmann schemes being kinetic [Simonis et al., 2020]—the number of discrete velocities is larger than the number of macroscopic equations. Therefore, the formal analyses for lattice Boltzmann schemes available in the literature try to bridge the gap between a kinetic and a macroscopic point of view by essentially relying on the quasi-equilibrium of the non-conserved variables. In particular, as far as the consistency with the macroscopic equations and the modified equations are concerned in the limit of small discretization parameters, two main approaches are at our disposal. The first one is based on the Chapman-Enskog expansion [Chapman and Cowling, 1990, Huang, 1987] from statistical mechanics, shaped to the context of lattice Boltzmann schemes, see for example [Chen and Doolen, 1998, Qian and Zhou, 2000]. The second approach features the so-called equivalent equations introduced by Dubois [Dubois, 2008, Dubois, 2022], consisting in performing a Taylor expansion of the scheme both for the conserved and non-conserved moments and progressively re-inject the developments order-by-order. This approach has proved to yield information in accordance with the numerical simulations, see [Dubois and Lallemand, 2009, Dubois and Lallemand, 2011]. Despite their proven empirical reliability and the fact that they yield the same results at the dominant orders (see [Dubois, 2019] for instance) these two strategies are both formal, especially for the computation of the truncation errors. Indeed, the Chapman-Enskog expansion relies on the introduction of two time variables with different scalings which are not present in the discrete lattice Boltzmann scheme. Moreover, in this approach and in the method of the equivalent equations, the values of the non-conserved variables are assumed to stem from the point-wise discretization of smooth functions, whose existence and smoothness cannot be guaranteed because they are absent from the target PDEs. Other approaches known in the literature are the asymptotic analysis under parabolic scaling deployed in [Junk and Yong, 2003, Junk et al., 2005, Junk and Yang, 2009] as well as the Maxwell iteration method [Yong et al., 2016, Zhao and

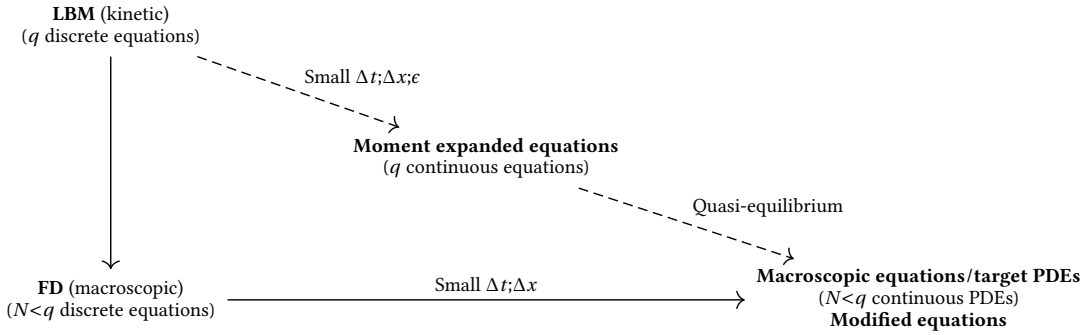


Figure 8.1: Different paths to recover the macroscopic equations and the modified equations. The formal approaches available in the literature [Chen and Doolen, 1998, Qian and Zhou, 2000, Dubois, 2008, Dubois, 2022, Yong et al., 2016] rely on the path marked with dashed arrows. They perform Taylor expansions for small discretization parameters and then utilize the quasi-equilibrium of the non-conserved moments to get rid of them. Our way of proceeding is marked with full arrows: we eliminate exactly the non-conserved moments at the discrete level as in Chapter 7 and we perform the usual analyses for Finite Difference schemes as in [Strikwerda, 2004, Allaire, 2007, Warming and Hyett, 1974, Carpentier et al., 1997].

Yong, 2017], which shares strong bonds with the equivalent equations method presented before. The previous list of formal analysis techniques does not aim at being exhaustive (the interested reader can refer to [Krüger et al., 2017]) and one should be aware that, despite efforts in this direction [Caiazzo et al., 2009], there is no consensus on which is the right method to use [Krüger et al., 2017]. A staple of all the previously mentioned approaches is that the expansion for the discretization parameters (time and space steps) tending to zero is performed on the kinetic numerical scheme, where both conserved and non-conserved variables are present. Eventually, the non-conserved variables are formally eliminated from the continuous formulation by scaling arguments, so to speak, using quasi-equilibrium. This corresponds to following the diagonal path on Figure 8.1.

Aims and structure of Chapter 8

To overcome the formal character of these approaches, in this Chapter 8, we develop the other path, namely the top-down movement followed by the left-right one on Figure 8.1. In particular, in order to fill the hollow between lattice Boltzmann schemes and traditional approaches known to numerical analysts, such as Finite Difference schemes, we have introduced—*cf.* Chapter 7—a formalism to recast any lattice Boltzmann scheme, regardless of its linearity, as a multi-step Finite Difference scheme solely on the conserved moments. This way of writing the scheme should be seen as a sort of one-way mathematical transform to pass from a kinetic standpoint to a macroscopic one in a purely discrete setting. The elimination of the non-conserved moments is carried out exactly on the discrete formulation by algebraic devices, thus independently from the time-space scaling. The price to pay for the non-conserved moments which do not relax exactly to their equilibrium value is the multi-step nature of the Finite Difference scheme. In Chapter 7 it has been crucial to be able to provide, thanks to a systematic mathematical approach, a precise description of the main ingredient needed to reduce the lattice Boltzmann scheme to a Finite Difference scheme, namely the characteristic polynomial of matrices of Finite Difference operators. We are therefore allowed to utilize this characteristic polynomial as a tool satisfying certain properties alone from the particular underlying lattice Boltzmann scheme. Quite the opposite, using the algorithm proposed by [Fučík and Straka, 2021], one is compelled to explicitly write down the corresponding Finite Difference scheme in order to perform the Taylor expansions to recover the target PDEs. In our case, the mathematical understanding that we *a priori* have on the corresponding (macroscopic) Finite Difference schemes, regardless of the (kinetic) lattice Boltzmann scheme they stand for, allows the following theoretical discussion. The theory of Finite Difference schemes features two important notions. One is the concept of truncation error ([Gustafsson et al., 1995, Defini-

tion 5.1.3] or [Allaire, 2007, Definition 2.2.4]), which is rigorous and is the basic ingredient to prove the celebrated Lax-Richtmyer equivalence theorem [Lax and Richtmyer, 1956]. The computations of the truncation error are perfectly justified because of the existence and smoothness results on the target PDEs (*e.g.* transport equation with smooth initial datum, Burgers equation with smooth non-decreasing initial datum, *etc.*). The second one is the concept of modified equation [Warming and Hyett, 1974, Carpentier et al., 1997], which is formal. The modified equations are those which the numerical scheme is "more consistent" with, compared to the target PDEs, and thus they yield essential but formal information on the behavior of the scheme. The modified equations cannot be fully justified even for Finite Difference schemes because they assume that smooth functions which equal the discrete solution of the scheme at the grid points exist.

The aim of Chapter 8 is to demonstrate results on consistency and modified equation for general lattice Boltzmann schemes thanks to the Finite Difference schemes without explicit computation of these ones. This allows to benefit from the rigorous framework of Finite Difference without the associated cost of an explicit computation. Moreover, we would like to validate the existing approaches to the consistency analysis, since they were only formal hitherto. Pursuing these objectives, Chapter 8 is structured as follows. In Section 8.1, the results of Section 7.5 are stated in a slightly different manner, facilitating the following analysis, thanks to the introduction of shift operators in time. Another important point which eases the computations is discussed in Section 8.2: the independence of the corresponding Finite Difference from the choice of relaxation parameters for the conserved moments, which allows to consider arbitrary values to our convenience. The main results of the consistency analysis are given in Section 8.3: under acoustic and diffusive scaling between time and space discretizations, we rigorously find the expression of the target PDEs approximated by any scheme and the associated truncation error. For the acoustic scaling, we also write the formal modified equations up to order two. Since the proof of these results is quite long, it is provided in the separate Section 8.4. Section 8.5 is devoted to hinting the links with some available approaches to find the modified equations of lattice Boltzmann schemes. Under acoustic scaling, the modified equations we obtain are the same as the ones by [Dubois, 2022] until second order. Moreover, rewriting the Maxwell iteration [Yong et al., 2016, Zhao and Yong, 2017] for general lattice Boltzmann schemes, allows us to show that both for the acoustic and diffusive scaling, the modified equations obtained through the corresponding Finite Difference scheme are the same as the ones from the Maxwell iteration at any order. Conclusions are drawn in Section 8.6.

Let us finally observe that our derivation of the truncation errors is rigorous—as the ones for Finite Difference schemes—and the formal modified equations rely on less unjustified assumptions than the existing approaches, for two main reasons. The first one is that Taylor expansions are applied to the conserved moments only, which also appear in the macroscopic equations. Therefore, one only postulates that the discrete conserved moments stem from the point-wise evaluation of smooth functions. The second one is that we solely rely on the link between time and space steps as the lattices are refined and which must be specified for any time-space numerical method.

Contents

8.1 More compact form of corresponding Finite Difference schemes

Although the asymptotic analysis we shall develop can be carried on the formulations from Proposition 7.5.1 and Proposition 7.5.2, we propose a different formalism based on shift operators in time. Having utilized both approaches, the advantage of this new standpoint—which shall be adopted in Chapter 8 and in the forthcoming material—is to easily deal with the asymptotic analysis of the coefficients of the characteristic polynomial and—more importantly—of the powers of the matrix $\mathbf{A}$ on the right hand side of (7.22) or (7.31). In particular, this allows for the straightforward generalization of the procedure above second-order. Furthermore, the links with other asymptotic analysis of lattice Boltzmann schemes from the literature—which we shall develop in Section 8.5—become noticeably more transparent. To this end, we introduce the following Definition.

Definition 8.1.1: Shift operator in time

Let $\mathsf{f} : \Delta t\mathbb{N} \to \mathbb{R}$ be any function defined on the time lattice, then the time shift operator z acts as

$$(\mathsf{z}\mathsf{f})(t) = \mathsf{f}(t + \Delta t), \qquad \forall t \in \Delta t\mathbb{N}.$$

With this, the scheme (7.21) can be recast under the fully-operatorial form:

$$(\mathsf{z}\boldsymbol{I} - \mathbf{A})\mathbf{m}(t, \boldsymbol{x}) = \mathbf{B}\mathbf{m}^{\mathrm{eq}}(t, \boldsymbol{x}), \qquad t \in \Delta t\mathbb{N}, \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d, \tag{8.1}$$

which corresponds to taking the Z-transform [Jury, 1964] of the scheme in the variable z. Here, the inverse of the resolvent associated with $\mathbf{A}$, namely $\mathsf{z}\boldsymbol{I} - \mathbf{A} \in \mathcal{M}_q(\mathbb{R}[\mathsf{z}] \otimes_{\mathbb{R}} \mathsf{D})$, where indeed $\mathbb{R}[\mathsf{z}] \otimes_{\mathbb{R}} \mathsf{D} \cong \mathbb{R}[\mathsf{z}, \mathsf{x}_1, \mathsf{x}_1^{-1}, \dots, \mathsf{x}_d, \mathsf{x}_d^{-1}]$, with $\otimes_{\mathbb{R}}$ indicating the tensor product of $\mathbb{R}$-algebras (see [Lang, 2002, Chapter 16] or [Kassel, 1995, Chapter 2]), forms a commutative ring. In the sequel, we shall drop the time and the space variables when not strictly needed for the sake of readability, because the system given by (8.1) is intrinsically time and space invariant thanks to the fact that the moment matrix $\boldsymbol{M}$ and the relaxation matrix $\boldsymbol{S}$ are assumed to be independent from the time and space variables, *cf.* Remark 7.5.6, and since we work on an unbounded domain, without considering the initial conditions.

We then have the equivalent of Proposition 7.5.1:

Proposition 8.1.1: Corresponding Finite Difference scheme for $N = 1$

Consider $N = 1$. Then the lattice Boltzmann scheme given by (7.21) or (8.1) corresponds to a multi-step explicit (macroscopic) Finite Difference scheme on the conserved moment m_1 under the form

$$\det(\mathsf{z}\boldsymbol{I} - \mathbf{A})\mathsf{m}_1 = \left(\mathrm{adj}(\mathsf{z}\boldsymbol{I} - \mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}}\right)_1, \tag{8.2}$$

where $\mathrm{adj}(\cdot)$ indicates the adjugate matrix,[a] also known as classical adjoint, which is the transpose of the cofactor matrix [Horn and Johnson, 2012].
Up to a temporal shift of the whole scheme, the corresponding multi-step explicit Finite Difference scheme by (8.2) equals the one from (7.22).

[a] It is worthwhile observing that the determinant and the adjugate matrix are defined for any square matrix with elements in a commutative ring.

Proof. The proof can be done starting from Proposition 7.5.1. Alternatively, using the fundamental relation between adjugate and determinant, see [Horn and Johnson, 2012, Chapter 0], which is a consequence of the Laplace formula, we have that for any $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$ where $\mathcal{R}$ is any commutative ring

$$\mathbf{C}\,\mathrm{adj}(\mathbf{C}) = \mathrm{adj}(\mathbf{C})\mathbf{C} = \det(\mathbf{C})\boldsymbol{I}. \tag{8.3}$$

Hence, multiplying (8.1) by $\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})$ yields $\det(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{m} = \mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}}$. Selecting the first row gives (8.2). □

Remark 8.1.1 (From kinetic to macroscopic). *We observe the following facts:*

- *The procedure can be reversed—when keeping all the lines in* $\det(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{m} = \mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}}$*—using a multiplication by* $\mathsf{z}\boldsymbol{I}-\mathbf{A}$ *and then dividing by the polynomial* $\det(\mathsf{z}\boldsymbol{I}-\mathbf{A})$*. In this way, one comes back to the lattice Boltzmann scheme by (8.1). This can be done as long as one does not select and store only the first row as in (8.2). Contrarily, if this selection is performed, the irreversible passage from the kinetic to the macroscopic formulation is accomplished. The non-conserved moments* $\mathsf{m}_2,\dots,\mathsf{m}_q$ *are no longer defined and they cannot be recovered from (8.2). This fact has been observed by [Dellacherie, 2014]: the same macroscopic Finite Difference scheme can correspond to distinct lattice Boltzmann schemes which can have different evolution equations for the non-conserved moments* $\mathsf{m}_2,\dots,\mathsf{m}_q$*. This is not surprising, since for a given monic polynomial, one can find an infinite number of matrices of which it is the characteristic polynomial.*
- *Though—as previously emphasized—the non-conserved moments are no longer present in the macroscopic Finite Difference scheme by (8.2), there is a residual shadow of their presence, namely the multi-step nature of the Finite Difference scheme, see Figure 7.2. In particular, each non-conserved moment* m_i *relaxing away from the equilibrium, namely with* $s_i \neq 1$*, for* $i \in [\![2,q]\!]$*, adds a time step to the corresponding Finite Difference scheme solely acting on the conserved moment* m_1*.*

Remark 8.1.2 (Adjugate and characteristic polynomial). *A time shift and a change of variable in (7.22) allows to express* $\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})$ *as a polynomial in* z *of degree* $q-1$ *computed from the characteristic polynomial. This relation is indeed classical and reads*

$$\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A}) = \sum_{k=0}^{q-1}\left(\sum_{r=0}^{q-1-k} \mathsf{c}_{k+r+1}\mathbf{A}^r\right)\mathsf{z}^k, \qquad \textit{where} \qquad \det(\mathsf{z}\boldsymbol{I}-\mathbf{A}) = \sum_{k=0}^{q}\mathsf{c}_k\mathsf{z}^k.$$

The time shift operator z *has just trivially taken the place of the general indeterminate of polynomials* X*.*

In the same way, we can restate and prove Proposition 7.5.2 using the new formalism.

Proposition 8.1.2: Corresponding Finite Difference scheme for $N \geq 1$

Consider $N \geq 1$. Then the lattice Boltzmann scheme given by (7.21) or (8.1) corresponds to a family of multi-step explicit (macroscopic) Finite Difference schemes on the conserved moments $\mathsf{m}_1,\dots,\mathsf{m}_N$. This is, for any $i \in [\![1,N]\!]$

$$\det(\mathsf{z}\boldsymbol{I}-\mathbf{A}_i)\mathsf{m}_i = \left(\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A}_i)\mathbf{A}_i^{\diamond}\mathbf{m}\right)_i + \left(\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A}_i)\mathbf{B}\mathbf{m}^{\mathrm{eq}}\right)_i, \tag{8.4}$$

where $\mathbf{A}_i = \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]}$ and $\mathbf{A}_i^{\diamond} = \mathbf{A}-\mathbf{A}_i$ by (7.29). Up to a temporal shift of the whole scheme, the corresponding multi-step explicit Finite Difference scheme by (8.4) equals the one from (7.31).

We could call the form of Finite Difference scheme from Proposition 7.5.2 and Proposition 8.1.2 "canonical" since we shall prove in Section 8.2 that it guarantees that the Finite Difference schemes do not depend on the choice of relaxation parameters for the conserved variables, which do not play any role in the original lattice Boltzmann scheme either, as previously discussed.

Remark 8.1.3 (Lack of scaling assumption). *The results in Proposition 7.5.1, Proposition 8.1.1, Proposition 7.5.2 and Proposition 8.1.2 are fully discrete and do not make any assumption on the particular scaling between the time-step* Δt *and the space-step* Δx*.*

The previous Remark 8.1.3 signifies that the corresponding Finite Difference schemes can be utilized to assess the consistency of the underlying lattice Boltzmann scheme with respect to the macroscopic equations for any particular scaling between time and space discretizations, as we will showcase in Section 8.3.

8.2 Independence of the choice of relaxation parameters for the conserved moments

In Chapter 1, we have observed that the choice of relaxation parameters for the conserved moments, namely $s_1, \dots, s_N$, does not change the lattice Boltzmann scheme (1.1). However, it could be argued that different choices for $s_1, \dots, s_N$ can affect the formulations of the corresponding Finite Difference schemes resulting from Proposition 8.1.2. We now show that, as one could hope, this is not the case for the Finite Difference schemes given by Proposition 8.1.2. To do this, we need the following result concerning the determinant of matrices under rank-one updates, whose proof is analogous to that in [Ding and Zhou, 2007] and plays an important role in several theoretical developments in Chapter 8, Chapter 9, and Chapter 10.

Lemma 8.2.1: Matrix determinant

Let $\mathcal{R}$ be a commutative ring, $\mathbf{C} \in \mathcal{M}_q(\mathcal{R})$ and $\mathbf{u}, \mathbf{v} \in \mathcal{R}^q$, then $\det(\mathbf{C} + \mathbf{u} \otimes \mathbf{v}) = \det(\mathbf{C}) + \mathbf{v}^{\mathrm{t}} \mathrm{adj}(\mathbf{C}) \mathbf{u}$.

We have the following result.

Proposition 8.2.1: Independence of the corresponding Finite Difference schemes on the values of $s_1, \dots, s_N$

The multi-step explicit macroscopic Finite Difference schemes given by (8.4) in Proposition 8.1.2 do not depend on the choice of $s_1, \dots, s_N$, the relaxation parameters of the conserved moments.

Proof. Fix the indices of the conserved moment $i \in [\![1, N]\!]$. Let us decompose $\mathbf{B}$, the part of the lattice Boltzmann scheme dealing with the equilibria, as follows: $\mathbf{B} = \mathbf{b}_i \otimes \boldsymbol{e}_i + \mathbf{B}|_{s_i=0}$ where $\mathbf{b}_i = \mathbf{B}_{\cdot,i}$ is the i-th column of $\mathbf{B}$. On the one hand, the dependency of $\mathbf{B}$ on the choice of s_i is now fully contained in $\mathbf{b}_i$. On the other hand $\mathbf{B}|_{s_i=0}$ does not depend on it. The Finite Difference scheme from Proposition 8.1.2 can be therefore recast, upon rearranging and using well-known properties of the external product $\otimes$, as

$$\left(\det(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i) - \boldsymbol{e}_i^{\mathrm{t}} \mathrm{adj}(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i)\mathbf{b}_i\right) \mathrm{m}_i = \left(\mathrm{adj}(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i)\mathbf{A}_i^{\diamond}\mathbf{m}\right)_i + \left(\mathrm{adj}(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i)\mathbf{B}|_{s_i=0}\mathbf{m}^{\mathrm{eq}}\right)_i . \tag{8.5}$$

The left hand side does not depend on s_j for $j \in [\![1, N]\!] \setminus \{i\}$ by construction of $\mathbf{A}_i$ and $\mathbf{b}_i$. On the other hand, the right hand side does not depend on s_j for $j \in [\![1, N]\!] \setminus \{i\}$, because $(\mathbf{A}_i^{\diamond})_{\cdot,j} + (\mathbf{B}|_{s_i=0})_{\cdot,j} = (\mathbf{A}_i^{\diamond}|_{s_j=0})_{\cdot,j}$, where we have used (1.1) and (1.3). We are left to discuss the possible dependency of (8.5) on s_i. By Lemma 8.2.1, we deduce that (8.5) now reads

$$\det(\mathrm{z}\boldsymbol{I} - (\mathbf{A}_i + \mathbf{b}_i \otimes \boldsymbol{e}_i)) m_i = \left(\mathrm{adj}(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i)\mathbf{A}_i^{\diamond}\boldsymbol{m}\right)_i + \left(\mathrm{adj}(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i)\mathbf{B}|_{s_i=0}\mathbf{m}^{\mathrm{eq}}\right)_i . \tag{8.6}$$

Observe that $\mathbf{A}_i + \mathbf{b}_i \otimes \boldsymbol{e}_i = \mathbf{A}_i|_{s_i=0}$, thus the left hand side of (8.6) does not not depend on s_i. The right hand side of (8.6) is independent of s_i because $\mathbf{A}_i^{\diamond}$ does not depend on it and since the i-th row of $\mathrm{adj}(\mathrm{z}\boldsymbol{I} - \mathbf{A}_i)$—the transpose of the cofactor matrix of $\mathrm{z}\boldsymbol{I} - \mathbf{A}_i$—cannot depend on s_i, because only the i-th column of $\mathrm{z}\boldsymbol{I} - \mathbf{A}_i$ depends on s_i. This concludes the proof. □

We have thus shown that the Finite Difference schemes from Proposition 8.1.2 are the same regardless of the choice of relaxation parameters for the conserved moments and so that we are allowed to take them equal to zero or any other value of specific convenience without loss of generality. In particular, the choice of taking $s_i = 0$ for $i \in [\![1, N]\!]$—*cf.* [Février, 2014]—offers interesting simplifications in the computations to come, in a way that shall be clearer by looking at the details. Moreover, this choice has the advantage of showing which moments are conserved at a glance.

8.3 Target equations, truncation errors and modified equations

Everything is in place to start the standard consistency analysis [Strikwerda, 2004, Allaire, 2007] and computation of the modified equations [Warming and Hyett, 1974, Carpentier et al., 1997] of Finite Difference schemes. We stress the fact that we aim at studying these features for (8.2) and (8.4) without explicitly writing these schemes down.

8.3.1 Assumptions, notations and scalings

We start from the assumptions allowing us to identify each term once developing in formal power series of Δx, *i.e.* performing Taylor expansions. Observe that for any time-space numerical scheme at hand, the time step Δt and the space step Δx are linked (scaling) when the grids are refined. Therefore, we decide to take Δx as discretization parameter tending to zero. Specific bonds between these two parameters will be given in the following pages.

Assumptions 8.3.1: General assumptions

Assume that the change of basis $\boldsymbol{M}$ and the relaxation matrix $\boldsymbol{S}$ do not depend on Δx.

We also introduce the spaces of differential operators which shall be obtained by taking the limit $\Delta x \to 0$ as well as other tightly associated concepts.

Definition 8.3.1: Time-space differential operators and related concepts

We define.

- The commutative ring of time-space differential operators:

$$\mathcal{D} := \mathbb{R}[\partial_t] \otimes_{\mathbb{R}} \mathbb{R}[\partial_{x_1}, \dots, \partial_{x_d}] \cong \mathbb{R}[\partial_t, \partial_{x_1}, \dots, \partial_{x_d}]$$

- We consider the commutative ring of formal power series [Niven, 1969, Monforte and Kauers, 2013] $\mathcal{S} := \mathcal{D}[\![\Delta x]\!]$.
- For any $\delta = \sum_{h=0}^{h=+\infty} \Delta x^h \delta^{(h)} \in \mathcal{S}$, we indicate $\delta = O(\Delta x^{h_\circ})$ for some $h_\circ \in \mathbb{N}$ if $\delta^{(h)} = 0$ for $h \in [\![0, h_\circ - 1]\!]$ and $\delta^{(h_\circ)} \neq 0$. The integer $h_\circ$ is called "order" of the formal power series δ, see [Roman, 2005, Chapter 1].
- Finally, let $\mathsf{d} \in \mathbb{R}[\mathsf{z}] \otimes_{\mathbb{R}} \mathsf{D}$ and $\delta \in \mathcal{S}$, then we indicate $\mathsf{d} \asymp \delta$, called "asymptotic equivalence" of d and δ, if for any smooth function of the time and space variables $f : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}$, we have

$$(\mathsf{d} f)(t, \boldsymbol{x}) = \sum_{h=0}^{+\infty} \Delta x^h (\delta^{(h)} f)(t, \boldsymbol{x}), \qquad \forall (t, \boldsymbol{x}) \in \mathbb{R} \times \mathbb{R}^d, \qquad \text{as} \qquad \Delta x \to 0.$$

  This definition is made possible by the embedding $\Delta t \mathbb{N} \times \Delta x \mathbb{Z}^d \subset \mathbb{R}_+ \times \mathbb{R}^d \subset \mathbb{R} \times \mathbb{R}^d$.

The previous $O(\cdot)$ notation and the notion of asymptotic equivalence are effortlessly extended to vectors and matrices in an entry-wise fashion. It shall be common and harmless not to distinguish between $\mathcal{M}_q(\mathcal{S})$ and $(\mathcal{M}_q(\mathcal{D}))[\![\Delta x]\!]$.

The momentum-velocity operator matrix $\boldsymbol{\mathcal{G}} \in \mathcal{M}_q(\mathcal{D})$, introduced by [Dubois, 2022] with slightly different notations, is defined as follows. It is indeed closely linked to the moment-stream matrix $\mathsf{T} \in \mathcal{M}_q(\mathsf{D})$ that we have previously introduced in Section 7.5.

Definition 8.3.2: Momentum-velocity operator matrix

The momentum-velocity operator matrix made up of first-order differential operators in space is given by

$$\boldsymbol{\mathcal{G}} := \boldsymbol{M} \sum_{|\mathbf{n}|=1} \operatorname{diag}(\boldsymbol{c}_1^{\mathbf{n}}, \dots, \boldsymbol{c}_q^{\mathbf{n}}) \partial_{\boldsymbol{x}}^{\mathbf{n}} \boldsymbol{M}^{-1} \in \mathcal{M}_q(\mathcal{D}), \tag{8.7}$$

where the multi-index notation is employed.

This momentum-velocity operator matrix can be partitioned in four blocks with different meanings according to the different nature (conserved or not) of the corresponding moments, as for [Dubois, 2022, Equation (8)].

As previously announced, one needs to specify the used scaling between Δt and Δx in order to perform the consistency analysis and also to recover the modified equations. We start by the acoustic scaling, see for example [Dubois, 2008, Dubois, 2022, Yong et al., 2016], where $\Delta t \propto \Delta x$.

Assumptions 8.3.2: Acoustic scaling

The assumptions when considering schemes with the acoustic scaling are:

1. $\lambda > 0$ is a fixed real number as $\Delta x \to 0$.
2. The moments at equilibrium $\boldsymbol{m}^{\mathrm{eq}}$ are fixed as $\Delta x \to 0$.

For the diffusive scaling, see [Zhao and Yong, 2017, Zhang et al., 2019], where $\Delta t \propto \Delta x^2$, we have:

Assumptions 8.3.3: Diffusive scaling

The assumptions when considering schemes with the diffusive scaling are:

1. $\lambda = \mu/\Delta x$ where $\mu > 0$ is a fixed real number as $\Delta x \to 0$.
2. $\mathcal{G}_{ij} = 0$ for $i, j \in [\![1, N]\!]$.
3. $m_i^{\mathrm{eq}} = \Delta x \hat{m}_i^{\mathrm{eq}}$ where $\hat{m}_i^{\mathrm{eq}}$ are fixed, for $i \in \hat{\Omega} := \{j \in [\![1, q]\!] \, : \, \mathcal{G}_{\ell j} \neq 0 \text{ for some } \ell \in [\![1, N]\!]\}$, as $\Delta x \to 0$.
4. m_i^{eq} for $i \notin \hat{\Omega}$ are fixed as $\Delta x \to 0$.

Remark 8.3.1 (Need for assumptions). *These assumptions are needed to state the general results to come. However, there are examples in the literature [Boghosian et al., 2018] where they are violated, in particular because the relaxation parameters depend on Δx. This does not prevent from writing the corresponding Finite Difference schemes (8.2) or (8.4) for the lattice Boltzmann scheme at hand and then recover their modified equations, but introduces a difficulty to directly obtain the modified equations without explicitly write (8.2) or (8.4) down.*

8.3.2 Theorems

We are now ready to state and then prove the main results of Chapter 8. The Taylor expansions are applied to the solution of the corresponding Finite Difference schemes, where non-conserved moments have been removed yielding purely macroscopic discrete equations.

Theorem 8.3.1: Consistency under acoustic scaling

Under Assumptions 8.3.1, Assumptions 8.3.2 and in the limit $\Delta x \to 0$, the corresponding macroscopic Finite Difference schemes given by Proposition 8.1.1 or Proposition 8.1.2 are consistent with the target PDEs

$$\partial_t \tilde{m}_i + \lambda \sum_{j=1}^{N} \mathcal{G}_{ij} \tilde{m}_j + \lambda \sum_{j=N+1}^{q} \mathcal{G}_{ij} m_j^{\mathrm{eq}}(\tilde{m}_1, \dots, \tilde{m}_N) = 0, \tag{8.8}$$

for $i \in [\![1, N]\!]$. For smooth solutions $\tilde{m}_1, \dots, \tilde{m}_N : \mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$ of (8.8), the truncation error is given by

$$\tau_i = \lambda \Delta x \sum_{j=N+1}^{q} \left(\frac{1}{s_j} - \frac{1}{2}\right) \mathcal{G}_{ij} \Bigg(\sum_{\ell=1}^{N} \mathcal{G}_{j\ell} \tilde{m}_\ell + \sum_{\ell=N+1}^{q} \mathcal{G}_{j\ell} m_\ell^{\mathrm{eq}}(\tilde{m}_1, \dots, \tilde{m}_N) \\ - \frac{1}{\lambda} \sum_{\ell=1}^{N} \frac{\mathrm{d} m_j^{\mathrm{eq}}}{\mathrm{d} m_\ell}(\tilde{m}_1, \dots, \tilde{m}_N) \gamma_{1,\ell}(\tilde{m}_1, \dots, \tilde{m}_N) \Bigg) + O(\Delta x^2),$$

where $\gamma_{1,i}(\tilde{m}_1, \dots, \tilde{m}_N) := \lambda \sum_{j=1}^{j=N} \mathcal{G}_{ij} \tilde{m}_j + \lambda \sum_{j=N+1}^{j=q} \mathcal{G}_{ij} m_j^{\mathrm{eq}}(\tilde{m}_1, \dots, \tilde{m}_N)$. Therefore, the modified equations up to second order read

$$\partial_t m_i + \gamma_{1,i}(m_1,\dots,m_N) - \lambda\Delta x \sum_{j=N+1}^{q}\left(\frac{1}{s_j}-\frac{1}{2}\right)\mathcal{G}_{ij}\Bigg(\sum_{\ell=1}^{N}\mathcal{G}_{j\ell}m_\ell + \sum_{\ell=N+1}^{q}\mathcal{G}_{j\ell}m_\ell^{\mathrm{eq}}(m_1,\dots,m_N) - \frac{1}{\lambda}\sum_{\ell=1}^{N}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_\ell}(m_1,\dots,m_N)\gamma_{1,\ell}(m_1,\dots,m_N)\Bigg) = O(\Delta x^2).$$

The first term in $\gamma_{1,i}$ represents the derivatives of fluxes of the conserved variables, which are necessarily linear, while the second one represents the derivatives of the fluxes given by the equilibria of the non-conserved moments, which can be non-linear. In the numerical diffusion terms, the so-called Hénon's parameters [Hénon, 1987] of type $1/s_j - 1/2$ appear. These terms are proportional to Δx. This is not surprising, since the only way of having a stable explicit Finite Difference scheme to simulate the heat equation is to consider a diffusion coefficient proportional to Δx under acoustic scaling or equivalently to take a diffusive scaling with fixed diffusion coefficient, in order to constrain the speed of propagation of information to remain finite in the limit $\Delta x \to 0$, see for instance [Strikwerda, 2004, Theorem 6.3.1].

Let us provide two examples for specific lattice Boltzmann schemes taken from the literature and employed with the acoustic scaling.

Example 8.3.1 ($\mathrm{D_1Q_3}$ with one conserved moment - acoustic scaling)**.** *We consider the* $\mathrm{D_1Q_3}$ *scheme presented in Section 1.5.2 with moment matrix given by* (1.6) *under dimensionless form and* $N = 1$*. We obtain*

$$\mathcal{G} = \begin{bmatrix} 0 & \partial_{x_1} & 0 \\ \frac{2}{3}\partial_{x_1} & 0 & \frac{1}{3}\partial_{x_1} \\ 0 & \partial_{x_1} & 0 \end{bmatrix}.$$

Theorem 8.3.1 immediately gives the modified equation for the acoustic scaling, which reads

$$\partial_t m_1 + \lambda\partial_{x_1} m_2^{\mathrm{eq}}(m_1) - \lambda\Delta x\left(\frac{1}{s_2}-\frac{1}{2}\right)\partial_{x_1}\left(\frac{2}{3}\partial_{x_1}m_1 + \frac{1}{3}\partial_{x_1}m_3^{\mathrm{eq}}(m_1) - \frac{\mathrm{d}m_2^{\mathrm{eq}}(m_1)}{\mathrm{d}m_1}\partial_{x_1}m_2^{\mathrm{eq}}(m_1)\right) = O(\Delta x^2).$$

Unsurprisingly, this coincides with (7.42) *when selecting linear equilibria, upon considering that we have taken a dimensionless moment matrix.*

Example 8.3.2 ($\mathrm{D_2Q_9}$ with three conserved moments - acoustic scaling)**.** *We consider the* $\mathrm{D_2Q_9}$ *scheme presented in Section 1.5.4 with* $N = 3$ *and dimensionless moment matrix* $\boldsymbol{M}$*, hence*

$$\mathcal{G} = \begin{bmatrix} 0 & \partial_{x_1} & \partial_{x_2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{2}{3}\partial_{x_1} & 0 & 0 & \frac{1}{6}\partial_{x_1} & 0 & 0 & 0 & \frac{1}{2}\partial_{x_1} & \partial_{x_2} \\ \frac{2}{3}\partial_{x_2} & 0 & 0 & \frac{1}{6}\partial_{x_2} & 0 & 0 & 0 & -\frac{1}{2}\partial_{x_2} & \partial_{x_1} \\ 0 & \partial_{x_1} & \partial_{x_2} & 0 & \partial_{x_1} & \partial_{x_2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3}\partial_{x_1} & 0 & 0 & \frac{1}{3}\partial_{x_1} & -\partial_{x_1} & \partial_{x_2} \\ 0 & 0 & 0 & \frac{1}{3}\partial_{x_2} & 0 & 0 & \frac{1}{3}\partial_{x_2} & \partial_{x_2} & \partial_{x_1} \\ 0 & 0 & 0 & 0 & \partial_{x_1} & \partial_{x_2} & 0 & 0 & 0 \\ 0 & \frac{1}{3}\partial_{x_1} & -\frac{1}{3}\partial_{x_2} & 0 & -\frac{1}{3}\partial_{x_1} & \frac{1}{3}\partial_{x_2} & 0 & 0 & 0 \\ 0 & \frac{2}{3}\partial_{x_2} & \frac{2}{3}\partial_{x_1} & 0 & \frac{1}{3}\partial_{x_2} & \frac{1}{3}\partial_{x_1} & 0 & 0 & 0 \end{bmatrix}.$$

The equilibria defining the modified equations under acoustic scaling at second-order are taken as those in Section 2.8.2.2, with the exception that in (2.56)*, the lattice velocity is set to* $\lambda = 1$*. Observe that* m_7^{eq} *do not need to be specified for this order of the analysis. It is well-known [Février, 2014, Dubois, 2022] that the* $O(\Delta x)$ *terms for the second and third modified equations contain spurious third-order contributions in* m_2, m_3*. We shall neglect these terms considering that they are small (low-speed flow). Furthermore, we consider that* m_1 *varies slowly as far as the* $O(\Delta x)$ *term is concerned, thus we neglect its derivatives. Moreover, we take* $s_9 = s_8$*, see [Lallemand and Luo, 2000, Dubois, 2022]. Under these assumptions, the modified equations from Theorem 8.3.1 read*

$$\partial_t m_1 + \partial_{x_1}\overline{m}_2 + \partial_{x_2}\overline{m}_3 = O(\Delta x^2),$$

$$\partial_t \overline{m}_2 + \partial_{x_1}\left(\frac{\overline{m}_2^2}{m_1} + \frac{\lambda^2}{3} m_1\right) + \partial_{x_2}\left(\frac{\overline{m}_2 \overline{m}_3}{m_1}\right) - \frac{\lambda}{3}\Delta x\left(\partial_{x_1}\left(2\left(\frac{1}{s_8} - \frac{1}{2}\right)\partial_{x_1}\overline{m}_2 + \left(\frac{1}{s_4} - \frac{1}{s_8}\right)(\partial_{x_1}\overline{m}_2 + \partial_{x_2}\overline{m}_3)\right) + \partial_{x_2}\left(\left(\frac{1}{s_8} - \frac{1}{2}\right)(\partial_{x_2}\overline{m}_2 + \partial_{x_1}\overline{m}_3)\right)\right) = O(\Delta x^2),$$

$$\partial_t \overline{m}_3 + \partial_{x_1}\left(\frac{\overline{m}_2 \overline{m}_3}{m_1}\right) + \partial_{x_2}\left(\frac{\overline{m}_3^2}{m_1} + \frac{\lambda^2}{3} m_1\right) - \frac{\lambda}{3}\Delta x\left(\partial_{x_1}\left(\left(\frac{1}{s_8} - \frac{1}{2}\right)(\partial_{x_2}\overline{m}_2 + \partial_{x_1}\overline{m}_3)\right) + \partial_{x_2}\left(2\left(\frac{1}{s_8} - \frac{1}{2}\right)\partial_{x_2}\overline{m}_3 + \left(\frac{1}{s_4} - \frac{1}{s_8}\right)(\partial_{x_1}\overline{m}_2 + \partial_{x_2}\overline{m}_3)\right)\right) = O(\Delta x^2),$$

where we have used $\overline{m}_2 := \lambda m_2$ and $\overline{m}_3 := \lambda m_3$. The first equation enforces the conservation of the density m_1 in the Euler system, discretized with a second-order scheme. The momentum along the first axis (respectively, second) is $\overline{m}_2$ (respectively, $\overline{m}_3$). The second equation represents—at leading order—the conservation of momentum along the first axis in the Euler system. The pressure law is linear and prescribes that the pressure is equal to $\lambda^2 m_1/3$, hence the speed of the sound is $\lambda/\sqrt{3}$. The numerical diffusion at order $O(\Delta x)$ is what makes up the terms that are usually recognized (except for the previously described pressure) as the stress tensor from the Navier-Stokes system. Recalling that we have assumed slow variations of m_1 (weakly compressible flow), we have a first bulk viscosity (also known as shear or dynamic viscosity) which equals $\mu = \lambda\Delta x(1/s_8 - 1/2)m_1/3$ (not linked μ in Assumptions 8.3.3) and a second bulk viscosity (also known as volume viscosity) given by $\kappa = \lambda\Delta x(3(1/s_4 - 1/2) - (1/s_8 - 1/2))m_1/9$. Hence, for this kind of system, the viscosity is modeled using numerical diffusion and is proportional to Δx, thus vanishing when going towards convergence. The same remarks hold for the last equation.

Theorem 8.3.2: Consistency under diffusive scaling

Under Assumptions 8.3.1, Assumptions 8.3.3 and in the limit $\Delta x \to 0$, the corresponding macroscopic Finite Difference schemes given by Proposition 8.1.1 or Proposition 8.1.2 are consistent with the target PDEs

$$\partial_t \tilde{m}_i + \mu \sum_{\substack{j=N+1 \\ j\in\hat{\Omega}}}^{q} \mathcal{G}_{ij}\hat{m}_j^{\mathrm{eq}}(\tilde{m}_1,\dots,\tilde{m}_N) - \mu \sum_{j=N+1}^{q}\left(\frac{1}{s_j} - \frac{1}{2}\right)\mathcal{G}_{ij}\left(\sum_{\ell=1}^{N}\mathcal{G}_{j\ell}\tilde{m}_\ell + \sum_{\substack{\ell=N+1 \\ \ell\notin\hat{\Omega}}}^{q}\mathcal{G}_{j\ell}m_\ell^{\mathrm{eq}}(\tilde{m}_1,\dots,\tilde{m}_N)\right) = 0, \quad (8.9)$$

for $i \in [\![1, N]\!]$. For smooth solutions $\tilde{m}_1,\dots,\tilde{m}_N : \mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$ of (8.9), the truncation error is given by $\tau_i = O(\Delta x)$.

We can *a posteriori* explain the meaning of some Assumptions 8.3.3 which were less clear before stating Theorem 8.3.2. The second assumption avoids to deal with terms which shall naturally appear at order $O(\Delta x)$ but which, since pertaining to the conserved moments, cannot be transformed into terms $O(\Delta x^2)$. Quite the opposite, the third assumption allows to rise to $O(\Delta x^2)$ those terms which contributed to the leading order in Theorem 8.3.1. This is achieved by a rescaling of the equilibria using $\hat{m}^{\mathrm{eq}}$. We therefore see that lattice Boltzmann schemes can be used to simulate non-linear transport/diffusion equations when using a diffusive scaling.

We also give two examples for specific lattice Boltzmann schemes considered under diffusive scaling. In order to reuse the scheme of Example 8.3.1 and Example 8.3.2, we have considered dimensionless moment matrices.

Example 8.3.3 ($\mathrm{D_1Q_3}$ with one conserved moment - diffusive scaling). *We come back to the setting of Example 8.3.1 except that we consider a diffusive scaling. Thus we have to take $m_2^{\mathrm{eq}}(\mathsf{m}_1) = \Delta x \hat{m}_2^{\mathrm{eq}}(\mathsf{m}_1)$ to comply with Assumptions 8.3.3. This yields the modified equation*

$$\partial_t m_1 + \mu\partial_{x_1}\hat{m}_2^{\mathrm{eq}}(m_1) - \mu\left(\frac{1}{s_2} - \frac{1}{2}\right)\partial_{x_1}\left(\frac{2}{3}\partial_{x_1}m_1 + \frac{1}{3}\partial_{x_1}m_3^{\mathrm{eq}}(m_1)\right) = O(\Delta x).$$

The scheme allows to simulate non-linear transport phenomena using $\hat{m}_2^{\mathrm{eq}}$ as well as linear and non-linear diffusion using m_3^{eq}.

Example 8.3.4 ($\mathrm{D_2Q_9}$ with one conserved moment - diffusive scaling). *We consider the same scheme as Example 8.3.2 except for the fact that only one conserved moment $N = 1$ is present and that the equilibria are general, with $m_2^{\mathrm{eq}}(\mathsf{m}_1) = \Delta x \hat{m}_2^{\mathrm{eq}}(\mathsf{m}_1)$ and $m_3^{\mathrm{eq}}(\mathsf{m}_1) = \Delta x \hat{m}_3^{\mathrm{eq}}(\mathsf{m}_1)$, to fulfill Assumptions 8.3.3. This is the setting introduced in [Zhang et al., 2019]. The modified equation reads*

$$\begin{aligned}
&\partial_t m_1 + \mu \partial_{x_1} \hat{m}_2^{\mathrm{eq}}(m_1) + \mu \partial_{x_2} \hat{m}_3^{\mathrm{eq}}(m_1) - \frac{2\mu}{3}\left(\frac{1}{s_2} - \frac{1}{2}\right)\partial_{x_1 x_1} m_1 - \frac{2\mu}{3}\left(\frac{1}{s_3} - \frac{1}{2}\right)\partial_{x_2 x_2} m_1 \\
&\quad - \frac{\mu}{6}\left(\left(\frac{1}{s_2} - \frac{1}{2}\right)\partial_{x_1 x_1} + \left(\frac{1}{s_3} - \frac{1}{2}\right)\partial_{x_2 x_2}\right) m_4^{\mathrm{eq}}(m_1) - \frac{\mu}{2}\left(\left(\frac{1}{s_2} - \frac{1}{2}\right)\partial_{x_1 x_1} - \left(\frac{1}{s_3} - \frac{1}{2}\right)\partial_{x_2 x_2}\right) m_8^{\mathrm{eq}}(m_1) \\
&\quad - \mu\left(\frac{1}{s_2} + \frac{1}{s_3} - 1\right)\partial_{x_1 x_2} m_9^{\mathrm{eq}}(m_1) = O(\Delta x).
\end{aligned}$$

Therefore, the scheme allows to simulate non-linear transport phenomena using $\hat{m}_2^{\mathrm{eq}}$ and $\hat{m}_3^{\mathrm{eq}}$ as well as linear and non-linear diffusion with crossed terms via *m_4^{eq}, m_8^{eq} and m_9^{eq}.*

In Chapter 8, we have deliberately neglected the behavior of the schemes close to the initial time $t = 0$. It is dictated by the choice of initial datum for the non-conserved moments, which is not unique for lattice Boltzmann schemes since $q > N$ but one only knows the N conserved moments at $t = 0$, being the initial datum of the target PDEs to be solved. This topic and its effects shall be discussed in Chapter 10.

Let us sketch the main ideas of the proofs of Theorem 8.3.1 and Theorem 8.3.2:

- The result of Proposition 7.5.2 has allowed to eliminate the non-conserved moments from the discrete scheme, thus has completed the step represented by a vertical arrow in Figure 8.1. Contrarily to the existing approaches, we do not need to (and we cannot, see Remark 8.1.1) estimate the Taylor expansions of the non-conserved moments.

- We benefit from the clever formulation from Proposition 8.1.2 instead of that of Proposition 7.5.2. Indeed, considering $\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i \asymp \mathsf{z}\boldsymbol{I} - \mathbf{A}_i$, we are allowed to write, for every $i \in [\![1, N]\!]$

$$\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i) m_i = \left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)\boldsymbol{\mathcal{A}}_i^{\diamond}\boldsymbol{m}\right)_i + \left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)\boldsymbol{\mathcal{B}}\boldsymbol{m}^{\mathrm{eq}}\right)_i,$$

  obtained applying the scheme to smooth functions $m_1, \dots, m_N : \mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$ and by replacing matrices with entries in the ring $\mathbb{R}[\mathsf{z}] \otimes_{\mathbb{R}} \mathsf{D}$ of discrete operators by their asymptotic equivalents in the ring $\mathcal{S}$. Here, for example, $\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i) \in \mathcal{S}$, and the expression perfectly makes sense because the determinant and the adjugate are well-defined polynomial functions of any square matrix on a commutative ring, like $\mathcal{S}$. Since the determinant and the adjugate are non-linear functions and thus mix different orders in the expansion $\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i$, if we want to recover a closed-form result at a given order of accuracy, we are compelled to utilize the Taylor expansions of the determinant and the adjugate. However, these expansions are well-known and can be computed at any order of accuracy.

  Quite the opposite, if we want to exploit the formulation stated in Proposition 7.5.2, we should characterize the asymptotic equivalents of any coefficient of the characteristic polynomial of $\mathbf{A}_i$ and then combine them with the asymptotic equivalents of the time shifts z alone and the terms on the right hand side of (7.31). Though this is actually feasible and we firstly did it, the computations are extremely involved[1] and very hard to generalize above second-order.

  This justifies the use of the formulation from Proposition 8.1.2 to achieve the step denoted by an horizontal arrow in Figure 8.1.

[1] Probably, a deeper mastery of the elementary symmetric polynomials, the Newton's identities, the Bell polynomials and the Feddeev-Leverrier algorithm could simplify the reasoning.

8.4 Proofs of the results in Section 8.3

The vast majority of rest of this work is devoted to the detailed proof of Theorem 8.3.1 for the scalar case $N = 1$. This choice has been adopted to keep the presentation and the involved notations as simple as possible. The idea behind the generalization to $N > 1$ is eventually given in Section 8.4.2 and is straightforward except for the more involved notations. The proof of Theorem 8.3.2 follows exactly the same path of Theorem 8.3.1 and is therefore omitted.

8.4.1 One conserved moment

Let us start by finding, for each shift operator from Definition 7.4.2, its asymptotically equivalent formal power series in Δx, see for instance [Yong et al., 2016, Dubois, 2022]. This is formalized by the following Lemma.

Lemma 8.4.1: Series expansion of a shift operator in space

Let $\boldsymbol{z} \in \mathbb{Z}^d$, then the associated shift operator in space $\mathsf{t}_{\boldsymbol{z}} \in \mathsf{T}$ is asymptotically equivalent, in the limit of $\Delta x \to 0$, to the formal power series of differential operators of the form

$$\mathsf{t}_{\boldsymbol{z}} \asymp \sum_{|\mathbf{n}| \geq 0} \frac{(-\Delta x)^{|\mathbf{n}|} \boldsymbol{z}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} \in \mathcal{S}.$$

Proof. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a smooth function of the spatial variable. Then performing a Taylor expansion for $\Delta x \to 0$ yields

$$(\mathsf{t}_{\boldsymbol{z}} f)(\boldsymbol{x}) = f(\boldsymbol{x} - \boldsymbol{z}\Delta x) = \sum_{|\mathbf{n}| \geq 0} \frac{(-\Delta x)^{|\mathbf{n}|} \boldsymbol{z}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} f(\boldsymbol{x}), \qquad \boldsymbol{x} \in \mathbb{R}^d.$$

□

The extension of Lemma 8.4.1 to any Finite Difference operator in D according to Definition 7.4.3 is done by linearity. With this in mind, recalling the definition of $\mathbf{T} \in \mathcal{M}_q(\mathsf{D})$, the moments-stream matrix and using Assumptions 8.3.1, we have that

$$\mathbf{T} := \boldsymbol{M} \mathrm{diag}(\mathsf{t}_{c_1}, \dots, \mathsf{t}_{c_q}) \boldsymbol{M}^{-1} \asymp \boldsymbol{M} \sum_{|\mathbf{n}| \geq 0} \frac{(-\Delta x)^{|\mathbf{n}|}}{\mathbf{n}!} \mathrm{diag}\left(\boldsymbol{c}_1^{\mathbf{n}}, \dots, \boldsymbol{c}_q^{\mathbf{n}}\right) \partial_{\boldsymbol{x}}^{\mathbf{n}} \boldsymbol{M}^{-1} =: \boldsymbol{\mathcal{T}} \in \mathcal{M}_q(\mathcal{S}). \tag{8.10}$$

Accordingly, we introduce $\boldsymbol{\mathcal{A}} := \boldsymbol{\mathcal{T}}(\boldsymbol{I} - \boldsymbol{S}) \in \mathcal{M}_q(\mathcal{S})$ and $\boldsymbol{\mathcal{B}} := \boldsymbol{\mathcal{T}}\boldsymbol{S} \in \mathcal{M}_q(\mathcal{S})$ such that $\boldsymbol{\mathcal{A}} \asymp \mathbf{A}$ and $\boldsymbol{\mathcal{B}} \asymp \mathbf{B}$. The tight bond between the momentum-velocity operator matrix $\boldsymbol{\mathcal{G}} \in \mathcal{M}_q(\mathcal{D})$ from [Dubois, 2022] and our moments-stream matrix $\mathbf{T} \in \mathcal{M}_q(\mathsf{D})$ and its asymptotic equivalent matrix $\boldsymbol{\mathcal{T}} \in \mathcal{M}_q(\mathcal{S})$ is given by the following Lemma.

Lemma 8.4.2: Link between $\mathcal{G}$ and $\mathcal{T}^{(h)}$

For any order $h \in \mathbb{N}$, the matrix $\boldsymbol{\mathcal{T}}^{(h)} \in \mathcal{M}_q(\mathcal{D})$ is linked to $\boldsymbol{\mathcal{G}} \in \mathcal{M}_q(\mathcal{D})$ by

$$\boldsymbol{\mathcal{T}}^{(h)} = \frac{(-1)^h}{h!} \boldsymbol{\mathcal{G}}^h.$$

Moreover, using the Assumptions 8.3.1, we also have

$$\boldsymbol{\mathcal{A}}^{(h)} = \frac{(-1)^h}{h!} \boldsymbol{\mathcal{G}}^h (\boldsymbol{I} - \boldsymbol{S}), \qquad \boldsymbol{\mathcal{B}}^{(h)} = \frac{(-1)^h}{h!} \boldsymbol{\mathcal{G}}^h \boldsymbol{S}.$$

Proof. By [Dubois, 2022, Equation (21)] or by direct comparison of (8.10) with (8.7), we have that $\mathbf{T} \asymp \boldsymbol{\mathcal{T}} = \exp(-\Delta x \boldsymbol{\mathcal{G}})$. The expansion of the exponential function yields the result. Using Assumptions 8.3.1, one obtains that $\boldsymbol{I} - \boldsymbol{S}$ and $\boldsymbol{S}$ do not perturb the orders of the expansion. □

As far as the time variable is concerned, we can complete by the development of the time shift operator z in order to provide the overall expansion of the inverse of the resolvent $z\boldsymbol{I} - \mathbf{A} \in \mathcal{M}_q(\mathbb{R}[z] \otimes_{\mathbb{R}} \mathsf{D})$.

Lemma 8.4.3: Expansion of the inverse of the resolvent of A

Under Assumptions 8.3.1, Assumptions 8.3.2 and in the limit of $\Delta x \to 0$, the inverse of the resolvent $z\boldsymbol{I} - \mathbf{A} \in \mathcal{M}_q(\mathbb{R}[z] \otimes_{\mathbb{R}} \mathsf{D})$ is asymptotically equivalent to $\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}} \in \mathcal{M}_q(\mathcal{S})$, where

$$\begin{aligned} z\boldsymbol{I} - \mathbf{A} \asymp \zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}} &= \sum_{h=0}^{+\infty} \frac{\Delta x^h}{h!} \left(\frac{1}{\lambda^h} \partial_t^h \boldsymbol{I} - (-1)^h \boldsymbol{\mathcal{G}}^h (\boldsymbol{I} - \boldsymbol{S}) \right) \\ &= \boldsymbol{S} + \Delta x \left(\frac{1}{\lambda} \partial_t \boldsymbol{I} + \boldsymbol{\mathcal{G}} (\boldsymbol{I} - \boldsymbol{S}) \right) + \frac{\Delta x^2}{2} \left(\frac{1}{\lambda^2} \partial_{tt} \boldsymbol{I} - \boldsymbol{\mathcal{G}}^2 (\boldsymbol{I} - \boldsymbol{S}) \right) + O(\Delta x^3). \end{aligned} \tag{8.11}$$

Proof. The standard Taylor expansion of z, using the assumption on the acoustic scaling, gives the claim. □

The consistency analysis of the Finite Difference schemes from Proposition 8.1.2 bis could be carried on infinite formal power series of differential operators $\mathcal{S}$ on the formulation

$$\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i) m_i = \left(\operatorname{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i) \boldsymbol{\mathcal{A}}_i^{\circ} \boldsymbol{m} \right)_i + \left(\operatorname{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i) \boldsymbol{\mathcal{B}} \boldsymbol{m}^{\mathrm{eq}} \right)_i, \tag{8.12}$$

for each $i \in [\![1, N]\!]$, because the determinant and the adjugate perfectly make sense for any square matrix on a commutative ring, like $\mathcal{S}$. However, in order to prove Theorem 8.3.1, where formal power series are truncated at a certain order, we shall need (8.11) from Lemma 8.4.3 as well as the Taylor expansions of the determinant and the adjugate matrix around a given matrix. Indeed, these are non-linear functions and thus mix different orders in the expansions $\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}} \in \mathcal{M}_q(\mathcal{S})$.

8.4.1.1 Determinant

We start by studying the expansion of the determinant up to second-order in the perturbation. For this, we need to characterize its derivatives. The expansion can be carried at higher order by employing the very same strategy.

Lemma 8.4.4: Derivatives and expansion of the determinant function

Let $\boldsymbol{\mathcal{C}} \in \mathrm{GL}_q(\mathcal{R})$ and $\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{E}} \in \mathcal{M}_q(\mathcal{R})$, where $\mathcal{R}$ is a commutative ring. Then the determinant function

$$\begin{aligned} \det \colon \mathcal{M}_q(\mathcal{R}) &\to \mathcal{R} \\ \boldsymbol{\mathcal{C}} &\mapsto \det(\boldsymbol{\mathcal{C}}), \end{aligned}$$

has the following derivatives.

$$\mathrm{D}_{\boldsymbol{\mathcal{C}}} (\det(\boldsymbol{\mathcal{C}})) (\boldsymbol{\mathcal{D}}) = \det(\boldsymbol{\mathcal{C}}) \operatorname{tr}(\boldsymbol{\mathcal{C}}^{-1} \boldsymbol{\mathcal{D}}), \tag{8.13}$$

$$\mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}} (\det(\boldsymbol{\mathcal{C}})) (\boldsymbol{\mathcal{D}}) (\boldsymbol{\mathcal{E}}) = \det(\boldsymbol{\mathcal{C}}) \left(\operatorname{tr}(\boldsymbol{\mathcal{C}}^{-1} \boldsymbol{\mathcal{E}}) \operatorname{tr}(\boldsymbol{\mathcal{C}}^{-1} \boldsymbol{\mathcal{D}}) - \operatorname{tr}(\boldsymbol{\mathcal{C}}^{-1} \boldsymbol{\mathcal{E}} \boldsymbol{\mathcal{C}}^{-1} \boldsymbol{\mathcal{D}}) \right), \tag{8.14}$$

The identity (8.13) is the celebrated and profound "Jacobi formula". Moreover, the second-order Taylor expansion of the determinant function reads

$$\det(\boldsymbol{\mathcal{C}} + \boldsymbol{\mathcal{D}}) = \det(\boldsymbol{\mathcal{C}}) + \mathrm{D}_{\boldsymbol{\mathcal{C}}} (\det(\boldsymbol{\mathcal{C}})) (\boldsymbol{\mathcal{D}}) + \frac{1}{2} \mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}} (\det(\boldsymbol{\mathcal{C}})) (\boldsymbol{\mathcal{D}}) (\boldsymbol{\mathcal{D}}) + O(\|\boldsymbol{\mathcal{D}}\|^3),$$

where the derivatives are given by (8.13) and (8.14).

Proof. The Jacobi formula (8.13) is a standard result, see [Horn and Johnson, 2012, Chapter 0] or [Zwillinger, 2018, Chapter 5]. Let us prove (8.14).

$$\mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}} (\det(\boldsymbol{\mathcal{C}})) (\boldsymbol{\mathcal{D}}) (\boldsymbol{\mathcal{E}}) := \mathrm{D}_{\boldsymbol{\mathcal{C}}} \left(\mathrm{D}_{\boldsymbol{\mathcal{C}}} (\det(\boldsymbol{\mathcal{C}})) (\boldsymbol{\mathcal{D}}) \right) (\boldsymbol{\mathcal{E}}) = \mathrm{D}_{\boldsymbol{\mathcal{C}}} \left(\det(\boldsymbol{\mathcal{C}}) \operatorname{tr}(\boldsymbol{\mathcal{C}}^{-1} \boldsymbol{\mathcal{D}}) \right) (\boldsymbol{\mathcal{E}}),$$

$$
\begin{aligned}
&= \mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{E}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}) + \det(\boldsymbol{\mathcal{C}})\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\right)(\boldsymbol{\mathcal{E}}),\\
&= \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}) + \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)(\boldsymbol{\mathcal{E}})),\\
&= \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}) - \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}),
\end{aligned}
$$

where we have used, in this order, the product rule for derivatives, the Jacobi formula (8.13), the linearity of the trace and the fact that $\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{D}}) = -\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\boldsymbol{\mathcal{C}}^{-1}$, see [Zwillinger, 2018, Chapter 5]. □

Remark 8.4.1 (On the invertibility assumption). *There exists a form of the Jacobi formula* (8.13) *for general* $\boldsymbol{\mathcal{C}} \in \mathcal{M}_q(\mathcal{R})$ *without assuming invertibility, under the form* $\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}}) = \mathrm{tr}(\mathrm{adj}(\boldsymbol{\mathcal{C}})\boldsymbol{\mathcal{D}})$. *This is equivalent to* (8.13), *since* (8.3) *holds. Nevertheless, we decided to state Lemma 8.4.4 using the invertibility assumption. This is done, as we shall see, without loss of generality by taking advantage of some invertible approximation of real matrices and allows to easily find the formulæ for higher order derivatives and expansions* via *basic differential calculus, as illustrated in the previous proof.*

In the sequel, we shall take $\mathcal{R} = \mathcal{S}$ and $\boldsymbol{\mathcal{C}} = \boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R}) \subset \mathrm{GL}_q(\mathcal{S})$ and $\boldsymbol{\mathcal{D}} = O(\Delta x) \in \mathcal{M}_q(\mathcal{S})$. To simplify the computations and relying on the findings of Section 8.2, we can consider $\boldsymbol{S}$ singular by having $s_1 = 0$. To avoid the difficulties linked with singular matrices, in the spirit of Remark 8.4.1, we take advantage of the fact that the derivatives of the determinant (and the determinant itself) around $\boldsymbol{\mathcal{C}}$ are smooth (indeed, polynomial) functions of $\boldsymbol{\mathcal{C}}$. Thus, we introduce the non-singular approximation $\boldsymbol{S}$ where $s_1 \neq 0$, which is such that $\boldsymbol{S} \to \boldsymbol{S}|_{s_1=0}$ as $s_1 \to 0$ for any matricial topology.

We are now ready to use the expansion given by Lemma 8.4.3 into the terms from Lemma 8.4.4 to find the leading order terms of the left hand side of (8.2), namely of $\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}) \in \mathcal{S}$. This is nothing but computing the Taylor series of composite functions (see the Faà di Bruno's formulæ[Johnson, 2002]) or the composition of formal series

$$
\begin{aligned}
\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}) = \det(\boldsymbol{S}) &+ \Delta x \mathrm{D}_{\boldsymbol{S}}\left(\det(\boldsymbol{S})\right)\left((\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(1)}\right)\\
&+ \Delta x^2\Big(\mathrm{D}_{\boldsymbol{S}}\left(\det(\boldsymbol{S})\right)\left((\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(2)}\right) + \frac{1}{2}\mathrm{D}_{\boldsymbol{S}\boldsymbol{S}}\left(\det(\boldsymbol{S})\right)\left((\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(1)}\right)\left((\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(1)}\right)\Big) + O(\Delta x^3).
\end{aligned}
$$

Since the product of the relaxation parameters for the non-conserved moments is a quantity which shall frequently appear in the computations to come, we fix a special notation for it, namely setting $\Pi := \prod_{i=2}^{i=q} s_i \neq 0$.

- One clearly has $\det(\boldsymbol{S}) = s_1\Pi$, because the matrix $\boldsymbol{S}$ is diagonal. Thus, the Taylor expansion of $\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})$ does not contain zero-order terms if $s_1 = 0$.
- Let $\boldsymbol{\mathcal{C}} = \boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R}) \subset \mathrm{GL}_q(\mathcal{S})$ and $\boldsymbol{\mathcal{D}} = \Delta x\left(\frac{1}{\lambda}\partial_t \boldsymbol{I} + \boldsymbol{\mathcal{G}}(\boldsymbol{I} - \boldsymbol{S})\right) + \frac{\Delta x^2}{2}\left(\frac{1}{\lambda^2}\partial_{tt}\boldsymbol{I} - \boldsymbol{\mathcal{G}}^2(\boldsymbol{I} - \boldsymbol{S})\right) + O(\Delta x^3) \in \mathcal{M}_q(\mathcal{S})$ from Lemma 8.4.3. Using (8.13) from Lemma 8.4.4 and performing elementary computations, we have

$$
\begin{aligned}
\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}}) &= \Delta x\Pi\left(\frac{1}{\lambda}\partial_t + (1 - s_1)\mathcal{G}_{11} + s_1\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1 - s_i)\mathcal{G}_{ii}\right)\right) + O(\Delta x^3) \qquad (8.15)\\
&+ \frac{\Delta x^2}{2}\Pi\left(\frac{1}{\lambda^2}\partial_{tt} - (1 - s_1)\mathcal{G}_{11}\mathcal{G}_{11} - (1 - s_1)\sum_{\ell=2}^{q}\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} + s_1\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda^2}\partial_{tt} - (1 - s_i)\sum_{\ell=1}^{q}\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right)\right).
\end{aligned}
$$

  We keep this expression without taking the limit in s_1, for future use. Taking the limit for $s_1 \to 0$ yields the derivative around the singular matrix $\boldsymbol{S}|_{s_1=0}$ instead of $\boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R})$ for $s_1 \neq 0$.

$$
\lim_{s_1 \to 0}\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}}) = \Delta x\Pi\left(\frac{1}{\lambda}\partial_t + \mathcal{G}_{11}\right) + \frac{\Delta x^2}{2}\Pi\left(\frac{1}{\lambda^2}\partial_{tt} - \mathcal{G}_{11}\mathcal{G}_{11} - \sum_{\ell=2}^{q}\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1}\right) + O(\Delta x^3). \qquad (8.16)
$$

  This gives all the first-order term and part of the second-order term in the series $\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})$.
- Let $\boldsymbol{\mathcal{C}} = \boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R}) \subset \mathrm{GL}_q(\mathcal{S})$ and $\boldsymbol{\mathcal{D}} = \Delta x\left(\frac{1}{\lambda}\partial_t \boldsymbol{I} + \boldsymbol{\mathcal{G}}(\boldsymbol{I} - \boldsymbol{S})\right) + O(\Delta x^2) \in \mathcal{M}_q(\mathcal{S})$ from Lemma 8.4.3. Using (8.14) from Lemma 8.4.4, we have, after some algebra

$$
\mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}})(\boldsymbol{\mathcal{D}}) = \Delta x^2\Pi\left(2\left(\frac{1}{\lambda}\partial_t + (1 - s_1)\mathcal{G}_{11}\right)\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1 - s_i)\mathcal{G}_{ii}\right)\right.
$$

$$+ s_1\Big(\sum_{i=2}^{q}\frac{1}{s_i}\Big(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\Big)\Big)^2 - 2(1-s_1)\sum_{\ell=2}^{q}\Big(\frac{1}{s_\ell}-1\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} - s_1\sum_{i=2}^{q}\frac{1}{s_i^2}\Big(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\Big)^2$$
$$- s_1\sum_{i=2}^{q}\sum_{\substack{\ell=2\\ \ell\neq i}}^{q}\Big(\frac{1}{s_i}-1\Big)\Big(\frac{1}{s_\ell}-1\Big)\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\Big) + O(\Delta x^3). \quad (8.17)$$

Once more, we take the limit for $s_1 \to 0$ in order to find the desired result on the remaining second-order terms in the development $\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})$

$$\lim_{s_1\to 0} \mathrm{D}_{\mathcal{C}\mathcal{C}}\left(\det(\mathcal{C})\right)(\mathcal{D})(\mathcal{D}) = 2\Delta x^2\Pi\Big(\frac{1}{\lambda^2}\partial_{tt}\sum_{\ell=2}^{q}\frac{1}{s_\ell} + \frac{1}{\lambda}\mathcal{G}_{11}\partial_t\sum_{\ell=2}^{q}\frac{1}{s_\ell} + \frac{1}{\lambda}\partial_t\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii}$$
$$+ \mathcal{G}_{11}\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii} - \sum_{\ell=2}^{q}\Big(\frac{1}{s_\ell}-1\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1}\Big) + O(\Delta x^3). \quad (8.18)$$

Putting (8.16) and (8.18) together in Lemma 8.4.4, with expansion around $\boldsymbol{S}$, allows to write $\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})$ up to third order. This is

$$\lim_{s_1\to 0}\det(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}) = \Delta x\Pi\Big(\frac{1}{\lambda}\partial_t + \mathcal{G}_{11}\Big) + \Delta x^2\Pi\Big(\frac{1}{\lambda^2}\Big(\frac{1}{2} + \sum_{\ell=2}^{q}\frac{1}{s_\ell}\Big)\partial_{tt} + \frac{1}{\lambda}\mathcal{G}_{11}\partial_t\sum_{\ell=2}^{q}\frac{1}{s_\ell} + \frac{1}{\lambda}\partial_t\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii}$$
$$- \frac{1}{2}\mathcal{G}_{11}\mathcal{G}_{11} - \sum_{\ell=2}^{q}\Big(\frac{1}{s_\ell}-\frac{1}{2}\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} + \mathcal{G}_{11}\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii}\Big) + O(\Delta x^3). \quad (8.19)$$

8.4.1.2 Adjugate

We now switch to the formal power series of the adjugate function of the inverse of the resolvent, in order to deal with the right hand side of the corresponding Finite Difference scheme given by (8.2). Let us start by characterizing its derivatives.

Lemma 8.4.5: Derivatives and expansion of the adjugate function

Let $\mathcal{C} \in \mathrm{GL}_q(\mathcal{R})$ and $\mathcal{D}, \mathcal{E} \in \mathcal{M}_q(\mathcal{R})$, where $\mathcal{R}$ is a commutative ring. Then the adjugate function

$$\begin{aligned}\mathrm{adj}\colon \mathcal{M}_q(\mathcal{R}) &\to \mathcal{M}_q(\mathcal{R})\\ \mathcal{C} &\mapsto \mathrm{adj}(\mathcal{C}),\end{aligned}$$

has the following derivatives.

$$\mathrm{D}_{\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D}) = \det(\mathcal{C})\left(\mathrm{tr}(\mathcal{C}^{-1}\mathcal{D})\boldsymbol{I} - \mathcal{C}^{-1}\mathcal{D}\right)\mathcal{C}^{-1}, \quad (8.20)$$

$$\mathrm{D}_{\mathcal{C}\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})(\mathcal{E}) = \det(\mathcal{C})\Big(\left(\mathrm{tr}(\mathcal{C}^{-1}\mathcal{E})\mathrm{tr}(\mathcal{C}^{-1}\mathcal{D}) - \mathrm{tr}(\mathcal{C}^{-1}\mathcal{E}\mathcal{C}^{-1}\mathcal{D})\right)\mathcal{C}^{-1}$$
$$+ \mathcal{C}^{-1}\left(\mathcal{E}\mathcal{C}^{-1}\mathcal{D} + \mathcal{D}\mathcal{C}^{-1}\mathcal{E} - \mathrm{tr}(\mathcal{C}^{-1}\mathcal{E})\mathcal{D} - \mathrm{tr}(\mathcal{C}^{-1}\mathcal{D})\mathcal{E}\right)\mathcal{C}^{-1}\Big). \quad (8.21)$$

Moreover, the second-order Taylor expansion of the adjugate function reads

$$\mathrm{adj}(\mathcal{C}+\mathcal{D}) = \mathrm{adj}(\mathcal{C}) + \mathrm{D}_{\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D}) + \frac{1}{2}\mathrm{D}_{\mathcal{C}\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})(\mathcal{D}) + O(\|\mathcal{D}\|^3),$$

where the derivatives are given by (8.20) and (8.21).

Proof. Since (8.3) holds and $\mathcal{C}$ is invertible, we have that $\mathrm{adj}(\mathcal{C}) = \det(\mathcal{C})\mathcal{C}^{-1}$. Therefore

$$\begin{aligned}\mathrm{D}_{\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D}) = \mathrm{D}_{\mathcal{C}}\left(\det(\mathcal{C})\mathcal{C}^{-1}\right)(\mathcal{D}) &= \mathrm{D}_{\mathcal{C}}\left(\det(\mathcal{C})\right)(\mathcal{D})\mathcal{C}^{-1} + \det(\mathcal{C})\mathrm{D}_{\mathcal{C}}\left(\mathcal{C}^{-1}\right)(\mathcal{D}),\\ &= \det(\mathcal{C})\mathrm{tr}(\mathcal{C}^{-1}\mathcal{D})\mathcal{C}^{-1} - \det(\mathcal{C})\mathcal{C}^{-1}\mathcal{D}\mathcal{C}^{-1},\end{aligned}$$

where we have used the rule for the derivative of a product, the Jacobi formula (8.13) and the identity $\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{D}}) = -\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\boldsymbol{\mathcal{C}}^{-1}$. For the second derivative, we have

$$\begin{aligned}
\mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}}\left(\mathrm{adj}(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}})(\boldsymbol{\mathcal{E}}) &:= \mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\mathrm{adj}(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}})\right)(\boldsymbol{\mathcal{E}}) = \mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{E}}),\\
&= \mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{E}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1} + \det(\boldsymbol{\mathcal{C}})\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{E}}),\\
&= \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1} + \det(\boldsymbol{\mathcal{C}})\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)(\boldsymbol{\mathcal{E}})\boldsymbol{\mathcal{C}}^{-1}\\
&\quad + \det(\boldsymbol{\mathcal{C}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{E}}),\\
&= \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1} + \det(\boldsymbol{\mathcal{C}})\left(\mathrm{tr}\left(\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{E}})\boldsymbol{\mathcal{D}}\right)\boldsymbol{I} - \mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{E}})\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1}\\
&\quad - \det(\boldsymbol{\mathcal{C}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{C}}^{-1},\\
&= \det(\boldsymbol{\mathcal{C}})\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1} - \det(\boldsymbol{\mathcal{C}})\left(\mathrm{tr}\left(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1}\\
&\quad - \det(\boldsymbol{\mathcal{C}})\left(\mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{I} - \boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\right)\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{C}}^{-1},
\end{aligned}$$

where we have used the rule for the derivative of a product, the Jacobi formula (8.13), the linearity of the derivative and the trace and the identity $\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\boldsymbol{\mathcal{C}}^{-1}\right)(\boldsymbol{\mathcal{D}}) = -\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\boldsymbol{\mathcal{C}}^{-1}$. Upon rearrangement, this yields the result. □

Remark 8.4.2. *We observe that, looking at (8.20) and (8.21) compared to (8.13) and (8.14), we have that*

$$\begin{aligned}
\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\mathrm{adj}(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}}) &= \mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}})\boldsymbol{\mathcal{C}}^{-1} - \det(\boldsymbol{\mathcal{C}})\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}}\boldsymbol{\mathcal{C}}^{-1},\\
\mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}}\left(\mathrm{adj}(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}})(\boldsymbol{\mathcal{E}}) &= \mathrm{D}_{\boldsymbol{\mathcal{C}}\boldsymbol{\mathcal{C}}}\left(\det(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}})(\boldsymbol{\mathcal{E}})\boldsymbol{\mathcal{C}}^{-1} + \det(\boldsymbol{\mathcal{C}})\boldsymbol{\mathcal{C}}^{-1}(\boldsymbol{\mathcal{E}}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}} + \boldsymbol{\mathcal{D}}\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}} - \mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{E}})\boldsymbol{\mathcal{D}} - \mathrm{tr}(\boldsymbol{\mathcal{C}}^{-1}\boldsymbol{\mathcal{D}})\boldsymbol{\mathcal{E}})\boldsymbol{\mathcal{C}}^{-1}.
\end{aligned}$$

This implies that we can reuse the computations we did for the determinant in the current treatment of the adjugate, as far as the first terms on the right hand sides are concerned. However, one must be careful that now they are multiplied by $\boldsymbol{\mathcal{C}}^{-1}$.

If we had stopped the developments at first order, we could have used the first-order perturbation theory of the adjugate matrix as provided by [Stewart, 1998, Theorem 2.1]. However, to the best of our knowledge, no second-order perturbation theory for this matrix is available in the literature, thus we have been compelled to independently develop it using differential calculus. Lemma 8.4.5 is thus a generalization of the results from [Stewart, 1998] and can therefore be used—beyond the application presented in this contribution—by researchers needing a second-order perturbation theory for the adjugate matrix.

Since we are ultimately interested, as one can notice from (8.2), in multiplying the formal power series $\mathrm{adj}(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}) \in \mathcal{M}_q(\mathcal{S})$ by $\boldsymbol{\mathcal{B}} \in \mathcal{M}_q(\mathcal{S})$ in a Cauchy-like fashion (the standard product of formal power series) and select the first row, we perform the computations only for the first row of $\mathrm{adj}(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})$.

- Using the definition of the adjugate matrix in combination with the Laplace formula or using the explicit formula for the adjugate of an upper triangular matrix, see [Horn and Johnson, 2012], we have

$$\mathrm{adj}(\boldsymbol{\mathcal{S}}) = \Pi\mathrm{diag}\left(1, \frac{s_1}{s_2}, \ldots, \frac{s_1}{s_q}\right), \qquad \text{thus} \qquad \lim_{s_1 \to 0}\mathrm{adj}(\boldsymbol{\mathcal{S}}) = \Pi\boldsymbol{e}_1 \otimes \boldsymbol{e}_1.$$

  Hence, contrarily to the determinant, the zero-order term in $\mathrm{adj}(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})$ is not zero for $s_1 = 0$ but a singular one-rank diagonal matrix.

- Let $\boldsymbol{\mathcal{C}} = \boldsymbol{\mathcal{S}} \in \mathrm{GL}_q(\mathbb{R}) \subset \mathrm{GL}_q(\mathcal{S})$ and $\boldsymbol{\mathcal{D}} = \Delta x\left(\frac{1}{\lambda}\partial_t\boldsymbol{I} + \boldsymbol{\mathcal{G}}(\boldsymbol{I} - \boldsymbol{\mathcal{S}})\right) + \frac{\Delta x^2}{2}\left(\frac{1}{\lambda^2}\partial_{tt}\boldsymbol{I} - \boldsymbol{\mathcal{G}}^2(\boldsymbol{I} - \boldsymbol{\mathcal{S}})\right) + O(\Delta x^3) \in \mathcal{M}_q(\mathcal{S})$ from Lemma 8.4.3. We utilize the previous computations from (8.15), as suggested in Remark 8.4.2, into (8.20).

$$\begin{aligned}
\mathrm{D}_{\boldsymbol{\mathcal{C}}}\left(\mathrm{adj}(\boldsymbol{\mathcal{C}})\right)(\boldsymbol{\mathcal{D}}) &= \left(\Delta x\Pi\left(\frac{1}{\lambda}\partial_t + (1 - s_1)\mathcal{G}_{11} + s_1\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1 - s_i)\mathcal{G}_{ii}\right)\right)\right.\\
&\quad \left. + \frac{\Delta x^2}{2}\Pi\left(\frac{1}{\lambda^2}\partial_{tt} - (1 - s_1)\mathcal{G}_{11}\mathcal{G}_{11} - (1 - s_1)\sum_{\ell=2}^{q}\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} + s_1\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda^2}\partial_{tt} - (1 - s_i)\sum_{\ell=1}^{q}\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right)\right)\right)\\
&\quad \times \mathrm{diag}\left(\frac{1}{s_1}, \frac{1}{s_2}, \ldots, \frac{1}{s_q}\right) - \mathrm{diag}\left(\frac{1}{s_1}, \frac{1}{s_2}, \ldots, \frac{1}{s_q}\right)\boldsymbol{\mathcal{D}}\mathrm{diag}\left(\frac{1}{s_1}, \frac{1}{s_2}, \ldots, \frac{1}{s_q}\right) + O(\Delta x^3).
\end{aligned}$$

In this case, we do not even have to take the limit for $s_1 \to 0$, since all the terms in s_1 cancel. Therefore, for the very first component, we get

$$\begin{aligned}\left(\mathrm{D}_{\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})\right)_{11} =& \Delta x \Pi\left(\frac{1}{\lambda}\partial_t \sum_{\ell=2}^{q}\frac{1}{s_\ell} + \sum_{i=2}^{q}\left(\frac{1}{s_i}-1\right)\mathcal{G}_{ii}\right)\\ &+\frac{\Delta x}{2}\Pi\left(\frac{1}{\lambda^2}\partial_{tt}\sum_{\ell=2}^{q}\frac{1}{s_\ell} - \sum_{i=2}^{q}\left(\frac{1}{s_i}-1\right)\sum_{\ell=1}^{q}\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right) + O(\Delta x^3). \end{aligned} \tag{8.22}$$

Now consider $j \in [\![2, q]\!]$, then

$$\left(\mathrm{D}_{\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})\right)_{1j} = -\Delta x \Pi\left(\frac{1}{s_j}-1\right)\mathcal{G}_{1j} + \frac{\Delta x^2}{2}\Pi\left(\frac{1}{s_j}-1\right)\left(\mathcal{G}_{11}\mathcal{G}_{1j} + \sum_{\ell=2}^{q}\mathcal{G}_{1\ell}\mathcal{G}_{\ell j}\right) + O(\Delta x^3). \tag{8.23}$$

This gives all the first-order terms on the first row of $\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})$ and part of the second-order terms.

- Let $\mathcal{C} = \boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R}) \subset \mathrm{GL}_q(\mathcal{S})$ and $\mathcal{D} = \Delta x\left(\frac{1}{\lambda}\partial_t \boldsymbol{I} + \boldsymbol{\mathcal{G}}(\boldsymbol{I} - \boldsymbol{S})\right) + O(\Delta x^2) \in \mathcal{M}_q(\mathcal{S})$ from Lemma 8.4.3. We reuse computations from (8.17) as well as (8.21).

$$\begin{aligned}\mathrm{D}_{\mathcal{C}\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})(\mathcal{D}) =& \left(\Delta x^2 \Pi\left(2\left(\frac{1}{\lambda}\partial_t + (1-s_1)\mathcal{G}_{11}\right)\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)\right.\right.\\ &+ s_1\left(\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)\right)^2 - 2(1-s_1)\sum_{\ell=2}^{q}\left(\frac{1}{s_\ell}-1\right)\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} - s_1\sum_{i=2}^{q}\frac{1}{s_i^2}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)^2\\ &\left.\left. - s_1\sum_{i=2}^{q}\sum_{\substack{\ell=2\\ \ell\neq i}}^{q}\left(\frac{1}{s_i}-1\right)\left(\frac{1}{s_\ell}-1\right)\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right)\right)\mathrm{diag}\left(\frac{1}{s_1}, \frac{1}{s_2}, \ldots, \frac{1}{s_q}\right)\\ &+ 2s_1\Pi\mathrm{diag}\left(\frac{1}{s_1}, \frac{1}{s_2}, \ldots, \frac{1}{s_q}\right)\left(\mathcal{D}\boldsymbol{S}^{-1}\mathcal{D} - \mathrm{tr}(\boldsymbol{S}^{-1}\mathcal{D})\mathcal{D}\right)\mathrm{diag}\left(\frac{1}{s_1}, \frac{1}{s_2}, \ldots, \frac{1}{s_q}\right) + O(\Delta x^3).\end{aligned}$$

Then we have, for the first matrix entry

$$\begin{aligned}\left(\mathrm{D}_{\mathcal{C}\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})(\mathcal{D})\right)_{11} =& \Delta x^2 \Pi\left(\left(\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)\right)^2 - \sum_{i=2}^{q}\frac{1}{s_i^2}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)^2\right.\\ &\left. - \sum_{i=2}^{q}\left(\frac{1}{s_i}-1\right)\sum_{\substack{\ell=2\\ \ell\neq i}}^{q}\left(\frac{1}{s_\ell}-1\right)\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right) + O(\Delta x^3),\end{aligned} \tag{8.24}$$

independent from s_1. For $j \in [\![2, q]\!]$

$$\begin{aligned}\left(\mathrm{D}_{\mathcal{C}\mathcal{C}}\left(\mathrm{adj}(\mathcal{C})\right)(\mathcal{D})(\mathcal{D})\right)_{1j} =& 2\Delta x^2 \Pi\left(\frac{1}{s_j}-1\right)\left(\frac{1}{s_j}\mathcal{G}_{1j}\left(\frac{1}{\lambda}\partial_t + (1-s_j)\mathcal{G}_{jj}\right) + \sum_{\substack{\ell=2\\ \ell\neq j}}^{q}\left(\frac{1}{s_\ell}-1\right)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j}\right.\\ &\left. - \mathcal{G}_{1j}\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)\right) + O(\Delta x^3).\end{aligned} \tag{8.25}$$

Using (8.22) and (8.24), we have that the first entry on the first row of $\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})$ is

$$\begin{aligned}\lim_{s_1 \to 0}\left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})\right)_{11} =& \Pi + \Delta x \Pi\left(\frac{1}{\lambda}\partial_t \sum_{\ell=2}^{q}\frac{1}{s_\ell} + \sum_{i=2}^{q}\left(\frac{1}{s_i}-1\right)\mathcal{G}_{ii}\right) + \frac{\Delta x^2}{2}\Pi\left(\frac{1}{\lambda^2}\partial_{tt}\sum_{\ell=2}^{q}\frac{1}{s_\ell} - \sum_{i=2}^{q}\left(\frac{1}{s_i}-1\right)\sum_{\ell=1}^{q}\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right.\\ &\left. + \left(\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)\right)^2 - \sum_{i=2}^{q}\frac{1}{s_i^2}\left(\frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii}\right)^2 - \sum_{i=2}^{q}\left(\frac{1}{s_i}-1\right)\sum_{\substack{\ell=2\\ \ell\neq i}}^{q}\left(\frac{1}{s_\ell}-1\right)\mathcal{G}_{i\ell}\mathcal{G}_{\ell i}\right) + O(\Delta x^3).\end{aligned} \tag{8.26}$$

Using (8.23) and (8.25), for any $j \in [\![2, q]\!]$, we write

$$\lim_{s_1 \to 0}\left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})\right)_{1j} = -\Delta x \Pi\left(\frac{1}{s_j}-1\right)\mathcal{G}_{1j} + \frac{\Delta x^2}{2}\Pi\left(\frac{1}{s_j}-1\right)\left(\mathcal{G}_{11}\mathcal{G}_{1j} + \sum_{\ell=2}^{q}\mathcal{G}_{1\ell}\mathcal{G}_{\ell j} + \frac{2}{s_j}\mathcal{G}_{1j}\left(\frac{1}{\lambda}\partial_t + (1-s_j)\mathcal{G}_{jj}\right)\right.$$

$$+2\sum_{\substack{\ell=2\\ \ell\neq j}}^{q}\Big(\frac{1}{s_\ell}-1\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j}-2\mathcal{G}_{1j}\sum_{i=2}^{q}\frac{1}{s_i}\Big(\frac{1}{\lambda}\partial_t+(1-s_i)\mathcal{G}_{ii}\Big)\Big)+O(\Delta x^3). \quad (8.27)$$

In general, we have written, for the first row, the leading terms in $\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})$. We shall take its product with $\boldsymbol{\mathcal{B}}$. Thus, one has

$$\begin{aligned}\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}} = \mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(0)}\boldsymbol{\mathcal{B}}^{(0)} &+ \Delta x\left(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(0)}\boldsymbol{\mathcal{B}}^{(1)}+\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(1)}\boldsymbol{\mathcal{B}}^{(0)}\right)\\ &+\Delta x^2\left(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(0)}\boldsymbol{\mathcal{B}}^{(2)}+\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(1)}\boldsymbol{\mathcal{B}}^{(1)}+\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(2)}\boldsymbol{\mathcal{B}}^{(0)}\right)+O(\Delta x^3), \quad (8.28)\end{aligned}$$

generating products of terms in the fashion of the Cauchy product. This completes the preliminary results needed to prove Theorem 8.3.1.

8.4.1.3 Overall computation

We now put all the previous calculations together to prove Theorem 8.3.1. As previously pointed out, we can assume, without loss of generality, that $s_1 = 0$, passing to the limit. This allows to deal with simpler expressions with less terms.

8.4.1.3.1 First-order equations To find the target PDE, it is sufficient to truncate all the formal power series at $O(\Delta x^2)$. In particular, using the fact that the first column of $\boldsymbol{\mathcal{B}}$ is zero for $s_1 = 0$, we have that $\lim_{s_1\to 0}(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}})_{11} = 0$. Observe that if the relaxation parameter corresponding to the conserved moment were not equal to zero, we would have $(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}})_{11} = O(1)$. Still the matrix $\boldsymbol{S}$ would not be singular, thus we would have some non vanishing zero-order term in $\det(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})$ to compensate the one from the adjugate.

For any $j \in [\![2, q]\!]$, using (8.27), Lemma 8.4.2 and (8.28), entails

$$\begin{aligned}\lim_{s_1\to 0}(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}})_{1j} &= \lim_{s_1\to 0}\Delta x\left(\left(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(0)}\boldsymbol{\mathcal{B}}^{(1)}\right)_{1j}+\left(\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})^{(1)}\boldsymbol{\mathcal{B}}^{(0)}\right)_{1j}\right)+O(\Delta x^2)\\ &= -\Delta x\Pi\mathcal{G}_{1j}+O(\Delta x^2).\end{aligned}$$

From (8.19), we obtain

$$\lim_{s_1\to 0}\det(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}}) = \Delta x\Pi\Big(\frac{1}{\lambda}\partial_t+\mathcal{G}_{11}\Big)+O(\Delta x^2),$$

thus we obtain the modified equation (whatever the choice of $s_1 \in \mathbb{R}$)

$$\Delta x\frac{\Pi}{\lambda}\Big(\partial_t m_1+\lambda\mathcal{G}_{11}m_1+\lambda\sum_{j=2}^{q}\mathcal{G}_{1j}m_j^{\mathrm{eq}}\Big)=O(\Delta x^2),$$

giving the desired result for $N = 1$ upon dividing by the constant Π. We explicitly see the target PDE. Observe that the term Π is never present in the computations by [Dubois, 2022] because they are done on the original lattice Boltzmann scheme (8.1) which has only one time step. For instance, in [Dubois, 2022], the multi-step nature of the problem, generated by the non-conserved moments relaxing away from the equilibrium, is damped at the very beginning of the procedure by performing the Taylor expansions of the scheme on the non-conserved variables and then plugging them into the expansions for the conserved moments.

Before clarifying the terms at the next order in the modified equation (for any m_1) or equivalently, finding the precise expression of the truncation error (for $m_1 \equiv \tilde{m}_1$ solution of the target PDE), let us utilize the previous equation to get rid of the time derivatives in the second order terms. This can be rigorously done if $m_1 \equiv \tilde{m}_1$, where $\tilde{m}_1$ is the smooth solution of the target PDE and yields the truncation error. For any m_1, this is formal because one assumes that differentiation preserves the asymptotic relations from the symbol $O(\cdot)$. This process constitutes the policy by [Dubois, 2008, Dubois, 2022] and is common to all the approaches (Chapman-Enskog, equivalent equation, Maxwell iteration, *etc.*) in order to find the value of the diffusion coefficients from the second-order terms. Moreover, this is classical for Finite Difference schemes, see [Warming and Hyett, 1974, Carpentier et al., 1997, Durran, 2013]. Very importantly, this is the reason why we can totally neglect to specify around which time we perform the Taylor expansions (this is not *a priori* trivial because schemes are multi-step). Notice that in this

case, where $N=1$, $\gamma_{1,\cdot}$ is a scalar, here denoted γ_1 for brevity.

$$\partial_t m_1 = -\lambda \mathcal{G}_{11} m_1 - \lambda \sum_{j=2}^{q} \mathcal{G}_{1j} m_j^{\mathrm{eq}} + O(\Delta x) = -\gamma_1 + O(\Delta x), \tag{8.29}$$

$$\partial_t \boldsymbol{m}^{\mathrm{eq}} = \frac{\mathrm{d}\boldsymbol{m}^{\mathrm{eq}}}{\mathrm{d}m_1}\partial_t m_1 = -\frac{\mathrm{d}\boldsymbol{m}^{\mathrm{eq}}}{\mathrm{d}m_1}\left(\lambda \mathcal{G}_{11} m_1 + \lambda \sum_{j=2}^{q} \mathcal{G}_{1j} m_j^{\mathrm{eq}}\right) + O(\Delta x) = -\frac{\mathrm{d}\boldsymbol{m}^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1 + O(\Delta x), \tag{8.30}$$

$$\partial_{tt} m_1 = -\partial_t\left(\lambda \mathcal{G}_{11} m_1 + \lambda \sum_{j=2}^{q} \mathcal{G}_{1j} m_j^{\mathrm{eq}}\right) + O(\Delta x) = -\lambda \mathcal{G}_{11}\partial_t m_1 - \lambda \sum_{j=2}^{q} \mathcal{G}_{1j}\partial_t m_j^{\mathrm{eq}} + O(\Delta x), \tag{8.31}$$

$$= \lambda \mathcal{G}_{11}\gamma_1 + \lambda \sum_{j=2}^{q} \mathcal{G}_{1j}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1 + O(\Delta x) = \lambda^2 \mathcal{G}_{11}\mathcal{G}_{11} m_1 + \lambda^2 \mathcal{G}_{11}\sum_{j=2}^{q} \mathcal{G}_{1j} m_j^{\mathrm{eq}} + \lambda \sum_{j=2}^{q} \mathcal{G}_{1j}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1 + O(\Delta x). \tag{8.32}$$

These formal equalities are obtained by taking advantage either of the chain rule, since the moments at equilibrium are functions of the conserved moments, or of the re-injection of (8.29) by assuming that the differentiation preserves the asymptotic relations from the symbol $O(\cdot)$. These equalities become rigorous and lack of the $O(\Delta x)$ term if $m_1 \equiv \tilde{m}_1$, the smooth solution of the target PDE.

8.4.1.3.2 Second-order equations We can now go to the computation of the truncation error in Theorem 8.3.1, which is more involved due to the presence of more terms to estimate. To make the link with the findings of [Dubois, 2022], the increased complexity comes from the more intricate and entangled block structure of $\boldsymbol{\mathcal{G}}^2$. We have to treat the second-order term in (8.28), made up of three products. For any $j \in [\![2, q]\!]$ (once again, the first component vanishes for $s_1 = 0$)

- Using Lemma 8.4.2 and the zero-order expansion of the adjugate gives

$$\lim_{s_1 \to 0} \left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(0)} \boldsymbol{\mathcal{B}}^{(2)}\right)_{1j} = \frac{s_j \Pi}{2}\left(\mathcal{G}_{11}\mathcal{G}_{1j} + \sum_{\ell=2}^{q} \mathcal{G}_{1\ell}\mathcal{G}_{\ell j}\right).$$

- Using Lemma 8.4.2 with (8.26) and (8.27)

$$\lim_{s_1 \to 0} \left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(1)} \boldsymbol{\mathcal{B}}^{(1)}\right)_{1j} = -s_j \Pi\left(\frac{1}{\lambda}\mathcal{G}_{1j}\partial_t \sum_{\ell=2}^{q} \frac{1}{s_\ell} + \boldsymbol{\mathcal{G}}_{1j} \sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - 1\right)\mathcal{G}_{\ell\ell} - \sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - 1\right)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j}\right).$$

- Using Lemma 8.4.2 and (8.27)

$$\begin{aligned}\lim_{s_1 \to 0} \left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})^{(2)} \boldsymbol{\mathcal{B}}^{(0)}\right)_{1j} = \Pi(1 - s_j)\Bigg(&\frac{1}{2}\mathcal{G}_{11}\mathcal{G}_{1j} + \sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - \frac{1}{2}\right)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j} - \left(\frac{1}{s_j} - 1\right)\mathcal{G}_{1j}\mathcal{G}_{jj} \\ &+ \frac{1}{s_j}\mathcal{G}_{1j}\left(\frac{1}{\lambda}\partial_t + (1 - s_j)\mathcal{G}_{jj}\right) - \mathcal{G}_{1j}\sum_{i=2}^{q}\frac{1}{s_i}\left(\frac{1}{\lambda}\partial_t + (1 - s_i)\mathcal{G}_{ii}\right)\Bigg).\end{aligned}$$

Summing these three contributions and after some straightforward but tedious computations, the second-order term in (8.28) is given by

$$\lim_{s_1 \to 0} \left(\left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}}\right)^{(2)}\right)_{1j} = \Pi\left(\frac{1}{2}\mathcal{G}_{11}\mathcal{G}_{1j} + \sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - \frac{1}{2}\right)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j} - \frac{1}{\lambda}\left(1 + \sum_{\substack{\ell=2 \\ \ell \neq j}}^{q}\frac{1}{s_\ell}\right)\mathcal{G}_{1j}\partial_t - \mathcal{G}_{1j}\sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - 1\right)\mathcal{G}_{\ell\ell}\right).$$

Hence, using (8.30) to get rid of the time derivative of the equilibria, we have

$$\begin{aligned}\lim_{s_1 \to 0} \sum_{j=2}^{q} \left(\left(\mathrm{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}}\right)^{(2)}\right)_{1j} m_j^{\mathrm{eq}} = \Pi \sum_{j=2}^{q}\Bigg(&\frac{1}{2}\mathcal{G}_{11}\mathcal{G}_{1j} m_j^{\mathrm{eq}} + \sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - \frac{1}{2}\right)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j} m_j^{\mathrm{eq}} \\ &+ \frac{1}{\lambda}\left(1 + \sum_{\substack{\ell=2 \\ \ell \neq j}}^{q}\frac{1}{s_\ell}\right)\mathcal{G}_{1j}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1 - \mathcal{G}_{1j}\sum_{\ell=2}^{q}\left(\frac{1}{s_\ell} - 1\right)\mathcal{G}_{\ell\ell} m_j^{\mathrm{eq}}\Bigg) + O(\Delta x).\end{aligned}$$

Notice that in this result, a reminder of order $O(\Delta x)$ appears. Once again, if $m_1 \equiv \tilde{m}_1$, this reminder is not present and we would find part of the truncation error. Once more, using (8.29) and (8.32) to eliminate the time derivatives in the second-order terms from (8.19) gives

$$\begin{aligned}\lim_{s_1\to 0}(\det(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}}))^{(2)} m_1 &= \Pi\Big(\frac{1}{\lambda^2}\Big(\frac{1}{2}+\sum_{\ell=2}^{q}\frac{1}{s_\ell}\Big)\partial_{tt} m_1+\frac{1}{\lambda}\mathcal{G}_{11}\partial_t\sum_{\ell=2}^{q}\frac{1}{s_\ell}+\frac{1}{\lambda}\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii}\partial_t m_1\\ &\quad -\frac{1}{2}\mathcal{G}_{11}\mathcal{G}_{11} m_1-\sum_{\ell=2}^{q}\Big(\frac{1}{s_\ell}-\frac{1}{2}\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} m_1+\mathcal{G}_{11}\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii} m_1\Big)\\ &= \Pi\Big(\Big(\frac{1}{2}+\sum_{\ell=2}^{q}\frac{1}{s_\ell}\Big)\Big(\mathcal{G}_{11}\mathcal{G}_{11} m_1+\mathcal{G}_{11}\sum_{j=2}^{q}\mathcal{G}_{1j} m_j^{\mathrm{eq}}+\frac{1}{\lambda}\sum_{j=2}^{q}\mathcal{G}_{1j}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1\Big)\\ &\quad -\mathcal{G}_{11}\Big(\mathcal{G}_{11} m_1+\sum_{j=2}^{q}\mathcal{G}_{1j} m_j^{\mathrm{eq}}\Big)\sum_{\ell=2}^{q}\frac{1}{s_\ell}-\Big(\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii}\Big)\Big(\mathcal{G}_{11} m_1+\sum_{j=2}^{q}\mathcal{G}_{1j} m_j^{\mathrm{eq}}\Big)\\ &\quad -\frac{1}{2}\mathcal{G}_{11}\mathcal{G}_{11} m_1-\sum_{\ell=2}^{q}\Big(\frac{1}{s_\ell}-\frac{1}{2}\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell 1} m_1+\mathcal{G}_{11}\sum_{i=2}^{q}\Big(\frac{1}{s_i}-1\Big)\mathcal{G}_{ii} m_1\Big)+O(\Delta x).\end{aligned}$$

With this, after simplifications, we obtain the remaining term to master the second-order contributions in the modified equation of the Finite Differencescheme (8.2).

$$\begin{aligned}&(\det(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}}))^{(2)} m_1-\sum_{j=2}^{q}\left((\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}})^{(2)}\right)_{1j} m_j^{\mathrm{eq}}\\ &=-\Pi\Big(\sum_{j=2}^{q}\Big(\frac{1}{s_j}-\frac{1}{2}\Big)\mathcal{G}_{1j}\mathcal{G}_{j1} m_1+\sum_{j=2}^{q}\sum_{\ell=2}^{q}\Big(\frac{1}{s_\ell}-\frac{1}{2}\Big)\mathcal{G}_{1\ell}\mathcal{G}_{\ell j} m_j^{\mathrm{eq}}-\frac{1}{\lambda}\sum_{j=2}^{q}\Big(\frac{1}{s_j}-\frac{1}{2}\Big)\mathcal{G}_{1j}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1\Big)+O(\Delta x).\end{aligned}$$

To wrap up, these computations yield, together with the ones from Section 8.4.1.3.1, the expected result for $N=1$, which reads

$$\Delta x\frac{\Pi}{\lambda}\Big(\partial_t m_1+\gamma_1-\lambda\Delta x\sum_{j=2}^{q}\Big(\frac{1}{s_j}-\frac{1}{2}\Big)\mathcal{G}_{1j}\Big(\mathcal{G}_{j1} m_1+\sum_{\ell=2}^{q}\mathcal{G}_{j\ell} m_\ell^{\mathrm{eq}}-\frac{1}{\lambda}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_1}\gamma_1\Big)\Big)=O(\Delta x^3),$$

and thus proves Theorem 8.3.1.

8.4.2 Key ideas for the extension to several conserved moments

Here, we sketch the demonstration of Theorem 8.3.1 for any $N\geq 1$. For the sake of providing a quick and effective presentation of this matter, we limit ourselves to first-order in Δx. Select a conserved moment, which shall be indexed by $i\in[\![1,N]\!]$.

Remark 8.4.3. *The operation selecting rows and columns to yield $\mathbf{A}_i$ and $\mathbf{A}_i^{\diamond}$ from Proposition 8.1.2 does not change the orders of the expansions. This is, let $\mathbf{C}\in\mathcal{M}_q(\mathbb{R}[\mathsf{z}]\otimes_{\mathbb{R}}\mathsf{D})$ and $\boldsymbol{\mathcal{C}}=\sum_{h=0}^{h=+\infty}\Delta x^h\boldsymbol{\mathcal{C}}^{(h)}\in\mathcal{M}_q(\mathcal{S})$ such that $\mathbf{C}\asymp\boldsymbol{\mathcal{C}}$ and $I\subset[\![1,q]\!]$ a set of indices, then*

$$\mathbf{C}_I\asymp\left(\sum_{h=0}^{+\infty}\Delta x^h\boldsymbol{\mathcal{C}}^{(h)}\right)_I=\sum_{h=0}^{+\infty}\Delta x^h\left(\boldsymbol{\mathcal{C}}^{(h)}\right)_I.$$

Thus we have the analogous of Lemma 8.4.3, where $\mathsf{z}\boldsymbol{I}-\mathbf{A}_i\asymp\zeta\boldsymbol{I}-\boldsymbol{\mathcal{A}}_i$, with

$$\zeta\boldsymbol{I}-\boldsymbol{\mathcal{A}}_i=\sum_{h=0}^{+\infty}\frac{\Delta x^h}{h!}\left(\frac{1}{\lambda^h}\partial_t^h\boldsymbol{I}-(-1)^h\left(\boldsymbol{\mathcal{G}}^h(\boldsymbol{I}-\boldsymbol{S})\right)_{\{i\}\cup[\![N+1,q]\!]}\right).$$

The first two term in the expansion of the inverse of the resolvent are

$$(\zeta\boldsymbol{I}-\boldsymbol{\mathcal{A}}_i)^{(0)}=\mathrm{diag}(1,\dots,1,s_i,1,\dots,1,s_{N+1},\dots,s_q).$$

In the spirit of Remark 8.4.1, for the case $s_i=0$, we introduce a regularization with $s_i\neq 0$ and then we pass to the

limit. Moreover

$$(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)^{(1)} = \frac{1}{\lambda}\partial_t \boldsymbol{I} + \left[\begin{array}{ccc|c|ccc} 0\cdots 0 & 0 & 0\cdots 0 & 0 & \cdots & 0 \\ \vdots\ddots\vdots & \vdots & \vdots\ddots\vdots & \vdots & \ddots & \vdots \\ 0\cdots 0 & 0 & 0\cdots 0 & 0 & \cdots & 0 \\ \hline 0\cdots 0 & (1-s_i)\mathcal{G}_{ii} & 0\cdots 0 & (1-s_{N+1})\mathcal{G}_{i(N+1)} & \cdots & (1-s_q)\mathcal{G}_{iq} \\ \hline 0\cdots 0 & 0 & 0\cdots 0 & 0 & \cdots & 0 \\ \vdots\ddots\vdots & \vdots & \vdots\ddots\vdots & \vdots & \ddots & \vdots \\ 0\cdots 0 & 0 & 0\cdots 0 & 0 & \cdots & 0 \\ \hline\hline 0\cdots 0 & (1-s_i)\mathcal{G}_{(N+1)i} & 0\cdots 0 & (1-s_{N+1})\mathcal{G}_{(N+1)(N+1)} & \cdots & (1-s_q)\mathcal{G}_{(N+1)q} \\ \vdots\ddots\vdots & \vdots & \vdots\ddots\vdots & \vdots & \ddots & \vdots \\ 0\cdots 0 & (1-s_i)\mathcal{G}_{qi} & 0\cdots 0 & (1-s_{N+1})\mathcal{G}_{q(N+1)} & \cdots & (1-s_q)\mathcal{G}_{qq} \end{array}\right].$$

We thus have

- As for the case $N = 1$ treated in detail, we have that $\lim_{s_i\to 0}\det((\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)^{(0)}) = 0$. Using the formula for the adjugate of an upper triangular matrix, see [Horn and Johnson, 2012], we have $\lim_{s_i\to 0}\operatorname{adj}((\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)^{(0)}) = \Pi \boldsymbol{e}_i \otimes \boldsymbol{e}_i$, where in Section 8.4.2 $\Pi := \prod_{\ell=N+1}^{\ell=q} s_\ell$.

- Taking $\boldsymbol{\mathcal{C}} = (\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)^{(0)} \in \mathrm{GL}_q(\mathbb{R}) \subset \mathrm{GL}_q(\mathcal{S})$ and $\boldsymbol{\mathcal{D}} = \Delta x(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)^{(1)} + O(\Delta x^2) \in \mathcal{M}_q(\mathcal{S})$ in the Jacobi formula (8.13)

$$\begin{aligned}\lim_{s_i\to 0} \mathrm{D}_{\boldsymbol{\mathcal{C}}}(\det(\boldsymbol{\mathcal{C}}))(\boldsymbol{\mathcal{D}}) &= \lim_{s_i\to 0} \Delta x \Pi\left(\frac{s_i(N-1)}{\lambda}\partial_t + \frac{1}{\lambda}\partial_t + (1-s_i)\mathcal{G}_{ii} + \sum_{\ell=N+1}^{q}\frac{1}{s_\ell}\left(\frac{1}{\lambda}\partial_t + (1-s_\ell)\mathcal{G}_{\ell\ell}\right)\right) + O(\Delta x^2) \\ &= \Delta x \Pi\left(\frac{1}{\lambda}\partial_t + \mathcal{G}_{ii}\right) + O(\Delta x^2).\end{aligned}$$

To handle the term with the adjugate, observe that the first-order term is made up of the terms

$$(\operatorname{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i)\boldsymbol{\mathcal{A}}_i^{\diamond})^{(1)} = (\operatorname{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i))^{(0)}(\boldsymbol{\mathcal{A}}_i^{\diamond})^{(1)} + (\operatorname{adj}(\zeta \boldsymbol{I} - \boldsymbol{\mathcal{A}}_i))^{(1)}(\boldsymbol{\mathcal{A}}_i^{\diamond})^{(0)}, \tag{8.33}$$

and in particular, we are interested in the i-th line of this matrix. Because of the fact that $(\boldsymbol{\mathcal{A}}_i^{\diamond})^{(0)} = \operatorname{diag}(1 - s_1, \ldots, 1 - s_{i-1}, 0, 1 - s_{i+1}, \ldots, 1 - s_N, 0, \ldots, 0)$, the i-th line of the second term on the right hand side of (8.33)

is zero, thus we do not have do study it. For the remaining term, it can be easily seen that

$$(\boldsymbol{\mathcal{A}}_i^{\diamond})^{(1)} = -(\boldsymbol{I}-\boldsymbol{S}) \left[\begin{array}{ccc|c|ccc||ccc} \mathcal{G}_{11} & \cdots & \mathcal{G}_{1(i-1)} & \mathcal{G}_{1i} & \mathcal{G}_{1(i+1)} & \cdots & \mathcal{G}_{1N} & \mathcal{G}_{1(N+1)} & \cdots & \mathcal{G}_{1q} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{G}_{(i-1)1} & \cdots & \mathcal{G}_{(i-1)(i-1)} & \mathcal{G}_{(i-1)i} & \mathcal{G}_{(i-1)(i+1)} & \cdots & \mathcal{G}_{(i-1)N} & \mathcal{G}_{(i-1)(N+1)} & \cdots & \mathcal{G}_{(i-1)q} \\ \hline \mathcal{G}_{i1} & \cdots & \mathcal{G}_{i(i-1)} & 0 & \mathcal{G}_{i(i+1)} & \cdots & \mathcal{G}_{iN} & 0 & \cdots & 0 \\ \hline \mathcal{G}_{(i+1)1} & \cdots & \mathcal{G}_{(i+1)(i-1)} & \mathcal{G}_{(i+1)i} & \mathcal{G}_{(i+1)(i+1)} & \cdots & \mathcal{G}_{(i+1)N} & \mathcal{G}_{(i+1)(N+1)} & \cdots & \mathcal{G}_{(i+1)q} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{G}_{N1} & \cdots & \mathcal{G}_{N(i-1)} & \mathcal{G}_{Ni} & \mathcal{G}_{N(i+1)} & \cdots & \mathcal{G}_{NN} & \mathcal{G}_{N(N+1)} & \cdots & \mathcal{G}_{Nq} \\ \hline\hline \mathcal{G}_{(N+1)1} & \cdots & \mathcal{G}_{(N+1)(i-1)} & 0 & \mathcal{G}_{(N+1)(i+1)} & \cdots & \mathcal{G}_{(N+1)N} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{G}_{q1} & \cdots & \mathcal{G}_{q(i-1)} & 0 & \mathcal{G}_{q(i+1)} & \cdots & \mathcal{G}_{qN} & 0 & \cdots & 0 \end{array}\right],$$

thus we deduce that

$$\begin{aligned} &\big((\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}}_i)\boldsymbol{\mathcal{A}}_i^{\diamond})^{(1)}\big)_{i,\cdot} \\ &\qquad = -\Pi\big((1-s_1)\mathcal{G}_{i1},\dots,(1-s_{i-1})\mathcal{G}_{i(i-1)},0,(1-s_{i+1})\mathcal{G}_{i(i+1)},\dots,(1-s_N)\mathcal{G}_{iN},0,\dots,0\big). \end{aligned}$$

Dealing with the zero and first order term in $\mathrm{adj}(\zeta \boldsymbol{I}-\boldsymbol{\mathcal{A}}_i)\boldsymbol{\mathcal{B}}$ works the same than $N=1$, thus we do not repeat it. Moreover, these terms allow for the compensation of the dependence on the choice of the relaxation parameter of the other conserved moments $s_1,\cdots,s_{i-1},s_{i+1},\dots,s_N$ in the previous equation, thanks to (1.3).

Putting all the previously discussed facts together into the truncated (8.12) yields

$$\Delta x \frac{\Pi}{\lambda}\Big(\partial_t m_i + \lambda \mathcal{G}_{ii} m_i + \lambda \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathcal{G}_{ij} m_j + \lambda \sum_{j=N+1}^{q} \mathcal{G}_{ij} m_j^{\mathrm{eq}}\Big) = O(\Delta x^2),$$

which is the result from Theorem 8.3.1 for $N \geq 1$ at dominant order. The next order is demonstrated in the same way.

8.5 Links with the existing approaches

To finish Chapter 8, we briefly sketch the links with previous works on the target PDEs and modified equations like [Yong et al., 2016] and [Dubois, 2008, Dubois, 2022].

8.5.1 Equivalent equations [Dubois, 2008, Dubois, 2022]

Our result Theorem 8.3.1 coincides with the analogous result in [Dubois, 2022, Proposition 5] up to second order. The substantial difference is that we apply the Taylor expansions to the solution of the corresponding Finite Difference scheme given by Proposition 8.1.2, where non-conserved moments have been removed. We therefore reasonably conjecture that the obtained macroscopic equations coincide at any order. The mathematical justification of this conjecture shall be the object of future investigations.

The quasi-equilibrium, which is extensively used in [Dubois, 2022] can be somehow recovered in our previous discussion. Let $N=1$ to fix ideas. In the proof of Proposition 8.1.1, nothing prevents us from selecting, instead of

the first row, the $i \in [\![2, q]\!]$ row, corresponding to a non-conserved moment. This is

$$\det(z\boldsymbol{I} - \mathbf{A})\mathsf{m}_i = \left(\operatorname{adj}(z\boldsymbol{I} - \mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}}\right)_i. \tag{8.34}$$

Let us stress that even if this could seem to be a viable Finite Difference scheme for the non-conserved variable m_i, it is not independent from the conserved moment m_1 the equilibria depend on and furthermore, this formulation certainly depends on the choice of s_1, the relaxation parameter of the conserved moment. This is somehow unwanted since s_1 is *in fine* not present in the original lattice Boltzmann scheme. From the computations of Section 8.4, we see that

$$\det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}) = s_1\Pi + O(\Delta x), \qquad \operatorname{adj}(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}) = \Pi\operatorname{diag}\left(1, \frac{s_1}{s_2}, \dots, \frac{s_1}{s_q}\right) + O(\Delta x), \qquad \boldsymbol{\mathcal{B}} = \boldsymbol{S} + O(\Delta x).$$

Using the asymptotic equivalents truncated at leading order in (8.34) thus provides

$$s_1\Pi m_i + O(\Delta x) = s_1\Pi m_i^{\mathrm{eq}} + O(\Delta x), \qquad \text{hence also} \qquad m_i = m_i^{\mathrm{eq}} + O(\Delta x),$$

provided that $s_1 \neq 0$. This is the quasi-equilibrium of the non-conserved moments, which is re-injected in the lattice Boltzmann schemes to eliminate them in the procedure by [Dubois, 2022]. The previous procedure is formal because there is no guarantee that the discrete non-conserved moments m_i for $i \in [\![2, q]\!]$ in the scheme can be replaced by the point-wise values of a smooth function m_i, which existence is not guaranteed.

8.5.2 Maxwell iteration [Yong et al., 2016]

In [Yong et al., 2016], the computations have been carried only for the $\mathrm{D_2Q_9}$ scheme by [Lallemand and Luo, 2000] with $N = 3$, which we have presented in Example 8.3.2. In this part of our work, we are first going to develop the computations until third-order for any lattice Boltzmann scheme under acoustic scaling, *i.e.* Assumptions 8.3.2. Then, we are going to demonstrate that the modified equations obtained by the Maxwell iteration [Yong et al., 2016] and the one from the corresponding Finite Difference schemes are the same at any order, regardless of the time-space scaling. Here in Section 8.5.2, it is crucial to assume that $\boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R})$. Observe that this assumption ensures that $\det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})$ is a unit in the ring $\mathcal{S}$ or equivalently that $\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}$ belongs to $\mathrm{GL}_q(\mathcal{S})$. The Maxwell iteration [Yong et al., 2016] is constructed recursively by

$$\begin{cases} \boldsymbol{m}^{[0]} = \boldsymbol{m}^{\mathrm{eq}}, \\ \boldsymbol{m}^{[k]} = \boldsymbol{m}^{\mathrm{eq}} - \boldsymbol{S}^{-1}(\zeta\overline{\boldsymbol{\mathcal{T}}} - \boldsymbol{I})\boldsymbol{m}^{[k-1]}, \qquad k \geq 1, \end{cases}$$

thus it can be written, *via* a simple computation, at step $k \in \mathbb{N}$ as

$$\boldsymbol{m}^{[k]} = \left(\sum_{h=0}^{k} (-\boldsymbol{S}^{-1}(\zeta\overline{\boldsymbol{\mathcal{T}}} - \boldsymbol{I}))^h\right)\boldsymbol{m}^{\mathrm{eq}}, \tag{8.35}$$

where the quasi-equilibrium is encoded in the choice $\boldsymbol{m}^{[0]} = \boldsymbol{m}^{\mathrm{eq}}$ and where we have taken, as for (8.10)

$$\begin{aligned} \overline{\mathbf{T}} := \boldsymbol{M}\operatorname{diag}(\overline{\mathsf{t}_{c_1}}, \dots, \overline{\mathsf{t}_{c_q}})\boldsymbol{M}^{-1} &= \boldsymbol{M}\operatorname{diag}(\mathsf{t}_{-c_1}, \dots, \mathsf{t}_{-c_q})\boldsymbol{M}^{-1} \\ &\asymp \boldsymbol{M}\sum_{|\mathbf{n}|\geq 0} \frac{\Delta x^{|\mathbf{n}|}}{\mathbf{n}!}\operatorname{diag}\left(\boldsymbol{c}_1^{\mathbf{n}}, \dots, \boldsymbol{c}_q^{\mathbf{n}}\right)\partial_{\boldsymbol{x}}^{\mathbf{n}}\boldsymbol{M}^{-1} =: \overline{\boldsymbol{\mathcal{T}}} \in \mathcal{M}_q(\mathcal{S}). \end{aligned}$$

It is easy to see that $\boldsymbol{\mathcal{T}}\overline{\boldsymbol{\mathcal{T}}} = \overline{\boldsymbol{\mathcal{T}}}\boldsymbol{\mathcal{T}} = \boldsymbol{I}$ and moreover, in analogy with Lemma 8.4.3

$$\zeta\overline{\boldsymbol{\mathcal{T}}} - \boldsymbol{I} = \Delta x\left(\frac{1}{\lambda}\partial_t\boldsymbol{I} + \boldsymbol{\mathcal{G}}\right) + \frac{\Delta x^2}{2}\left(\frac{1}{\lambda^2}\partial_{tt}\boldsymbol{I} + \frac{2}{\lambda}\boldsymbol{\mathcal{G}}\partial_t + \boldsymbol{\mathcal{G}}^2\right) + O(\Delta x^3). \tag{8.36}$$

The Maxwell iteration works by assuming that $\boldsymbol{m} = \boldsymbol{m}^{[k]} + O(\Delta x^{k+1})$. Taking $k = 1$ in (8.35) and using (8.36), we have

$$\boldsymbol{m} = \boldsymbol{m}^{\mathrm{eq}} - \boldsymbol{S}^{-1}\Delta x\Big(\frac{1}{\lambda}\partial_t \boldsymbol{I} + \boldsymbol{\mathcal{G}}\Big)\boldsymbol{m}^{\mathrm{eq}} + O(\Delta x^2).$$

Let $i \in [\![1, N]\!]$, then taking advantage of (1.3)

$$m_i = m_i - \Delta x \frac{1}{s_i}\Big(\frac{1}{\lambda}\partial_t m_i + \sum_{j=1}^{N} \mathcal{G}_{ij} m_j + \sum_{j=N+1}^{q} \mathcal{G}_{1j} m_j^{\mathrm{eq}}\Big) + O(\Delta x^2),$$

which upon division, is the same result than Theorem 8.3.1. Going up to order two considering $k = 2$, we have

$$\begin{aligned}\boldsymbol{m} = \boldsymbol{m}^{\mathrm{eq}} &- \boldsymbol{S}^{-1}\Delta x\Big(\frac{1}{\lambda}\partial_t \boldsymbol{I} + \boldsymbol{\mathcal{G}}\Big)\boldsymbol{m}^{\mathrm{eq}} \\ &+ \frac{\Delta x^2}{2}\boldsymbol{S}^{-1}\Big(\frac{1}{\lambda^2}(2\boldsymbol{S}^{-1} - \boldsymbol{I})\partial_{tt} + \frac{2}{\lambda}(\boldsymbol{S}^{-1}\boldsymbol{\mathcal{G}} + \boldsymbol{\mathcal{G}}\boldsymbol{S}^{-1} - \boldsymbol{\mathcal{G}})\partial_t + \boldsymbol{\mathcal{G}}(2\boldsymbol{S}^{-1} - \boldsymbol{I})\boldsymbol{\mathcal{G}}\Big)\boldsymbol{m}^{\mathrm{eq}} + O(\Delta x^3).\end{aligned}$$

Once more, selecting the i-th row provides

$$\begin{aligned}m_i = m_i - \Delta x \frac{1}{s_i}\Big(\frac{1}{\lambda}\partial_t m_i + \frac{1}{\lambda}\gamma_{1,i}\Big) + \Delta x^2 \frac{1}{s_i}\Big(\frac{1}{\lambda^2}\Big(\frac{1}{s_i} - \frac{1}{2}\Big)\partial_{tt} m_i + \frac{1}{\lambda}\sum_{j=1}^{q}\Big(\frac{1}{s_i} + \frac{1}{s_j} - 1\Big)\mathcal{G}_{ij}\partial_t m_j^{\mathrm{eq}} \\ + \sum_{j=1}^{q}\sum_{\ell=1}^{q}\Big(\frac{1}{s_\ell} - \frac{1}{2}\Big)\mathcal{G}_{i\ell}\mathcal{G}_{\ell j} m_j^{\mathrm{eq}}\Big)\Big) + O(\Delta x^3).\end{aligned}$$

Using relations analogous to (8.29), (8.30) and (8.32) for $N \geq 1$, formally obtained by differentiating the result at the previous order , we finally obtain, after tedious but elementary computations

$$\begin{aligned}m_i = m_i - \frac{\Delta x}{\lambda s_i}\Big(\partial_t m_i + \gamma_{1,i} - \lambda\Delta x \sum_{j=N+1}^{q}\Big(\frac{1}{s_j} - \frac{1}{2}\Big)\mathcal{G}_{ij}\Big(\sum_{\ell=1}^{N}\mathcal{G}_{j\ell} m_\ell + \sum_{\ell=N+1}^{q}\mathcal{G}_{j\ell} m_\ell^{\mathrm{eq}} - \frac{1}{\lambda}\sum_{\ell=1}^{N}\frac{\mathrm{d}m_j^{\mathrm{eq}}}{\mathrm{d}m_\ell}\gamma_{1,\ell}\Big)\Big) \\ + O(\Delta x^3),\end{aligned}$$

which coincides with the result from Theorem 8.3.1. Therefore, up to order two, our approach yields results consistent with those from the procedure by [Yong et al., 2016].

To demonstrate that we recover the same result at any order for any scaling between time and space discretizations, let us assume $N = 1$. Then we have, using that $\boldsymbol{S} \in \mathrm{GL}_q(\mathbb{R})$, $\boldsymbol{\mathcal{T}}\overline{\boldsymbol{\mathcal{T}}} = \overline{\boldsymbol{\mathcal{T}}}\boldsymbol{\mathcal{T}} = \boldsymbol{I}$, the rule for the inverse of a product of matrix and the identity relative to geometric series in the context of formal power series, that

$$\begin{aligned}\boldsymbol{0} &= \det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\boldsymbol{m} - \mathrm{adj}(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\boldsymbol{\mathcal{B}}\boldsymbol{m}^{\mathrm{eq}} = \det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\left(\boldsymbol{m} - (\zeta\boldsymbol{I} - \boldsymbol{\mathcal{T}}(\boldsymbol{I} - \boldsymbol{S}))^{-1}\boldsymbol{\mathcal{T}}\boldsymbol{S}\boldsymbol{m}^{\mathrm{eq}}\right), \\ &= \det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\left(\boldsymbol{m} - (\boldsymbol{S}^{-1}\overline{\boldsymbol{\mathcal{T}}}(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{T}}(\boldsymbol{I} - \boldsymbol{S})))^{-1}\boldsymbol{m}^{\mathrm{eq}}\right) = \det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\left(\boldsymbol{m} - (\boldsymbol{I} + \boldsymbol{S}^{-1}(\zeta\overline{\boldsymbol{\mathcal{T}}} - \boldsymbol{I}))^{-1}\boldsymbol{m}^{\mathrm{eq}}\right), \\ &= \det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\left(\boldsymbol{m} - \left(\sum_{h=0}^{+\infty}(-\boldsymbol{S}^{-1}(\zeta\overline{\boldsymbol{\mathcal{T}}} - \boldsymbol{I}))^h\right)\boldsymbol{m}^{\mathrm{eq}}\right) = \det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}})\left(\boldsymbol{m} - \lim_{k\to+\infty}\boldsymbol{m}^{[k]}\right).\end{aligned}$$

Therefore the expansion of the Finite Difference scheme from Proposition 8.1.1 and the non-truncated Maxwell iteration method on the lattice Boltzmann scheme coincide up to a multiplication by a formal power series of time-space differential operators, *i.e.* $\det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}) \in \mathcal{S}$. *A priori*, the resulting modified equations are not the same, but since $\det(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}) = \det(\boldsymbol{S}) + O(\Delta x) = s_1\Pi + O(\Delta x)$, thus we "pay" only a constant factor we can divide by at dominant order, the modified equations at leading order are the same. Then, at each order, the result must be the same because we re-inject, in a recursive fashion, the solution truncated at the previous order to eliminate the higher-order time derivatives, see for instance (8.30) and (8.32). The fact that the modified equations recovered by the Maxwell iteration are the same than the ones from the corresponding Finite Difference scheme at any order provides an *a posteriori* justification of the Maxwell iteration. We also emphasize that using the Maxwell iteration to compute these equations is generally less involved in terms of computations than doing the same on

the corresponding Finite Difference schemes.

8.6 Conclusions of Chapter 8

In Chapter 8, we have rigorously derived the target PDEs for any lattice Boltzmann scheme under acoustic and diffusive scalings by restating it as a multi-step macroscopic Finite Difference scheme on the conserved moments. Moreover, the modified equations—which the schemes are "more consistent" with—have been found up to second order. These findings allow to utilize—upon studying the stability of the lattice Boltzmann scheme at hand—*cf.* Chapter 9—the Lax equivalence theorem [Lax and Richtmyer, 1956] to conclude on its convergence and order of convergence towards the solution of the target PDEs. Since the passage from the kinetic to the macroscopic standpoint is fully discrete, our analysis can handle any type of time-space scaling and be pushed forward to reach higher orders in the discretization parameters. Contrarily to the existing techniques, the quasi-equilibrium of the non-conserved moments in the limit of small discretization parameters or the introduction of several time scales in the problem are not the keys to eliminate the non-conserved variables from the macroscopic equations. The obtained results confirm, going beyond empirical evidence, that the formal Taylor expansion by [Dubois, 2008, Dubois, 2022] and the Maxwell iteration by [Yong et al., 2016] are well-grounded from the perspective of numerical analysts and traditional numerical methods for PDEs, such as Finite Difference. In particular, we have extended the Maxwell iteration [Yong et al., 2016] to any lattice Boltzmann scheme and shown that the modified equations found by this procedure are the same than the ones from the corresponding Finite Difference schemes, at any order. The general results that we have presented allow to immediately recover the modified equations without need for computing the corresponding Finite Difference schemes, which would be time consuming. This allows—for example—to easily consider families of schemes depending on some parameters and investigate the dependence of the modified equations on these factors.

An improvement of the present work could be the establishment of the equivalence between different analyses [Chen and Doolen, 1998, Qian and Zhou, 2000, Dubois, 2008, Dubois, 2022, Junk and Yong, 2003, Junk et al., 2005, Junk and Yang, 2009] for higher orders and ideally for any order. Even if more involved from the standpoint of computations, the extension can be easily done by considering derivatives of higher order for the determinant and adjugate functions, in the spirit of Lemma 8.4.4 and Lemma 8.4.5. In this work, all the computations have been done by hand but one could envision to seek some help from symbolic computations.

Chapter 9

Stability

General context and motivation

Stability is the second fundamental piece to achieve convergence for a numerical scheme. In fact, the Lax-Richtmyer theorem [Lax and Richtmyer, 1956] states that for linear Finite Difference schemes, when the solution of the target problem is smooth, consistency and stability are necessary and sufficient to obtain convergence. However, even if the stability of lattice Boltzmann has been studied for a long time, it remains problematic due to the lack of a well-established link with convergence and the complexity of the schemes.

State of the art

- The historical way of analyzing the stability of lattice Boltzmann methods is the L^2 analysis *à la von Neumann*, see [Benzi et al., 1992, Sterling and Chen, 1996, Lallemand and Luo, 2000, Graille, 2014, Février, 2014] to cite a few. The procedure relies on a linearization of the problem around an equilibrium state (for example, a base flow, or zero velocity [Benzi et al., 1992]), followed by the rewrite of the scheme using the Fourier transform and the study of the spectrum of the associated matrix. In particular, one checks that the modulus of the eigenvalues is smaller than one for any wave-number. The main limitations of this approach are the fact that it works only for linear problems and finding explicit stability constraints is difficult, especially when the number of discrete velocities q is large. Nevertheless, this remains the most common approach, at least for a numerical study of the stability regions.
- A more recent approach is built around a weighted L^2 stability analysis, see [Banda et al., 2006, Junk and Yong, 2009, Rheinländer, 2010] and applications to specific schemes [Junk and Yang, 2009, Junk and Yang, 2015]. The idea behind this approach is to decouple collision and stream, because the main difficulty in the analysis is the fact that they are not diagonal in the same basis. The key is to find a matrix which determines a weighted L^2 norm such that the stream phase remains an isometry and the collision phase be a contraction. More precisely, the matrix is picked in order to diagonalize the collision phase. In this way, one can evaluate the norm of their product by the one of each phase separately and conclude on the stability. Again, this notion of stability is intrinsically linear and does not take into account the joint role of collision and stream, which individual norm might be larger than one but the product of which may have a norm smaller than one, ensuring overall stability.
- Finally, even more recent developments are linked with the L^∞ stability, monotonicity, and related concepts, see [Dellacherie, 2014, Graille, 2014, Caetano et al., 2023, Dubois et al., 2020a]. In particular—while the other contribution still deal with a linear framework—the analysis by [Caetano et al., 2023] studies the D_1Q_2 from the perspective of monotonicity, like for Finite Volume schemes [Godlewski and Raviart, 2013], identifying an invariant compact set under suitable conditions and showing that this L^∞ stability leads to entropy inequalities and convergence towards the weak-entropic solution of a scalar conservation law. The limitation of this approach is that it works for very simple schemes: it is difficult to generalize it even to a D_1Q_3. Over-relaxation regimes, when the relaxation parameters is larger than one, are also stumbling

blocks and the sets of parameters where the scheme still possesses a maximum principle are missed by this theoretical analysis.

Aim and structure of Chapter 9

The aim of Chapter 9 is to relate the L^2 linear stability analysis *à la von Neumann* for lattice Boltzmann scheme, which is a rather heuristic approach, to the *von Neumann* stability analysis for Finite Difference scheme, see standard textbooks such as [Strikwerda, 2004, Gustafsson et al., 1995]. This allows us to justify the full legitimacy of the analysis on the original lattice Boltzmann scheme thanks to the corresponding Finite Difference schemes. Moreover, the aim is to avoid having to write the corresponding Finite Difference scheme, but once more rely on the *a priori* knowledge of it in order to perform the stability analyses on the original lattice Boltzmann scheme. In particular:

- For one conserved moment, namely $N = 1$, we demonstrate that the notion of *von Neumann* linear stability for a lattice Boltzmann scheme is perfectly equivalent to the one for its corresponding Finite Difference scheme given by Proposition 8.1.1.
- For several conserved moments, namely $N > 1$, the stability of the corresponding Finite Difference scheme given by Proposition 8.1.2 implies the stability of the original lattice Boltzmann scheme. The converse cannot be proved in full generality.

Chapter 9 is structured as follows. In Section 9.1, we introduce the linearization of the lattice Boltzmann scheme and the definition of *von Neumann* stability. We also explain why the spectrum of the scheme matrix plays, instead of its norm, a central role in the stability analysis. Then, in Section 9.2, we discuss the case of one conserved moment $N = 1$ by showing that the amplification polynomial of the corresponding Finite Difference scheme coincides with the characteristic polynomial of the scheme matrix of the lattice Boltzmann scheme. Therefore, the equivalence between stability definitions is automatically deduced. Then, we demonstrate that the transformation of the lattice Boltzmann scheme into a Finite Difference scheme is unique. In Section 9.3, we discuss the case of $N > 1$. We first introduce the needed construction of the *von Neumann* stability analysis for Finite Difference schemes dealing with systems of N equations. Then, we discuss the lack of uniqueness concerning the transformation of the lattice Boltzmann scheme into Finite Difference schemes. For the choice of corresponding Finite Difference scheme by Proposition 8.1.2, we clarify the link between the stability of the original lattice Boltzmann scheme and this scheme. Finally, in Section 9.4, which does not deal with stability, we use the previously described lack of uniqueness concerning the transformation of the lattice Boltzmann scheme into Finite Difference schemes when $N > 1$ to recover the results from [Ginzburg, 2009] concerning the link $\mathrm{D}_d\mathrm{Q}_{2W+1}$ with two-relaxation-times and magic parameters equal to $1/4$, which has already been studied in Section 7.6.3 and Example 7.6.2 solely for $N = 1$.

Contents

9.1 Linearization, *von Neumann* stability and role of the spectrum

9.1.1 Linearization and *von Neumann* stability

As previously pointed out, the *von Neumann* stability analysis, see [Benzi et al., 1992, Sterling and Chen, 1996, Lallemand and Luo, 2000], is quite likely the most employed notion of stability for lattice Boltzmann schemes. It consists in the linearization of the problem around an equilibrium state, followed by the rewrite of the scheme using the Fourier transform and the study of the spectrum of the derived matrix. Considering a linear or linearized lattice Boltzmann boils down to assuming that there exist vectors $\boldsymbol{\epsilon}_i \in \mathbb{R}^q$ for $i \in [\![1, N]\!]$ such that the equilibria are given by

$$\boldsymbol{m}^{\mathrm{eq}}(\mathsf{m}_1, \dots, \mathsf{m}_N) = \left(\sum_{i=1}^{N} \boldsymbol{\epsilon}_i \otimes \boldsymbol{e}_i \right) \mathbf{m}. \tag{9.1}$$

Hence, the lattice Boltzmann scheme (7.21) becomes

$$\mathbf{m}(t + \Delta t) = \underbrace{\left(\mathbf{A} + \mathbf{B} \sum_{i=1}^{N} \boldsymbol{\epsilon}_i \otimes \boldsymbol{e}_i \right)}_{=:\mathbf{E}} \mathbf{m}(t), \tag{9.2}$$

so that the action of the numerical scheme is wholly encoded in a scheme matrix $\mathbf{E} \in \mathcal{M}_q(\mathsf{D})$. Then, the *von Neumann* stability can be resumed in the following Definition [Février, 2014, Equation (1.57)]:

Definition 9.1.1: *Von Neumann* stability of a lattice Boltzmann scheme

Consider a linear or linearized lattice Boltzmann scheme, which therefore can be put under the form (9.2). We say that it is stable in the *von Neumann* sense if, for every $\boldsymbol{\theta} \in [-\pi, \pi]^d$, every $\hat{\mathsf{g}}_k(\boldsymbol{\theta}) \in \mathrm{sp}(\hat{\mathbf{E}}(\boldsymbol{\theta}))$ for $k \in [\![1, q]\!]$, where $\mathrm{sp}(\cdot)$ indicates the spectrum of a matrix, is such that:

$$|\hat{\mathsf{g}}_k(\boldsymbol{\theta})| \leq 1.$$

9.1.2 The role of the spectrum

The question which might be risen concerns the fact that in Definition 9.1.1, one looks at the spectrum of the matrix $\hat{\mathbf{E}}$ instead of at its L^2 norm $|||\hat{\mathbf{E}}||_2$, trying to establish the property $|||\hat{\mathbf{E}}||_2 \leq 1$. To understand this, consider the $\mathrm{D}_1\mathrm{Q}_2$ scheme of Section 7.1.1 in its linear version, with relaxation on the equilibrium $s_2 = 1$ and lattice velocity $\lambda = 1$, with "perfect" CFL condition $\boldsymbol{\epsilon}_1 \cdot \boldsymbol{e}_2 / \lambda = 1$. The corresponding Finite Difference scheme is trivially given by a shift of the datum, namely

$$\mathsf{m}_1(t + \Delta t, x) = \mathsf{m}_1(t, x - \Delta x),$$

which is trivially stable, especially in the ℓ^2-norm, since the associated amplification factor $e^{-i\theta}$ in the Fourier space lies on the unit circle: the scheme is an isometry. The amplification matrix of the lattice Boltzmann scheme in Fourier reads

$$\hat{\mathbf{E}}(\theta) = \begin{bmatrix} e^{-i\theta} & 0 \\ e^{-i\theta} & 0 \end{bmatrix}, \qquad \text{hence} \qquad \hat{\mathbf{E}}^*(\theta)\hat{\mathbf{E}}(\theta) = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix},$$

where $*$ indicates the conjugate transpose of a complex matrix. Clearly $\mathrm{sp}(\hat{\mathbf{E}}(\theta)) = \{e^{-i\theta}, 0\}$, hence the scheme is stable according to Definition 9.1.1, but

$$|||\hat{\mathbf{E}}(\theta)|||_2 = \max_{\sigma \in \mathrm{sp}(\hat{\mathbf{E}}^*(\theta)\hat{\mathbf{E}}(\theta))} \sqrt{|\sigma|} = \sqrt{2} > 1. \tag{9.3}$$

This means that the scheme operator $\mathbf{E}$ is not contractive for the L^2 norm. By considering the generalization of the stability inequality (7.44) to vectorial problems of size q but only with one time step, like the original lattice Boltzmann schemes are, the definition of stability—for a norm at hand—becomes [Strikwerda, 2004, Equation

(7.1.3)]: for any final time $T > 0$, there exists C_T such that

$$|||\mathbf{E}^{t/\Delta t}||| \leq C_T, \tag{9.4}$$

for every $t \in [\![1, n_T]\!]\Delta t$, regardless of the value of Δt (which indeed goes to zero). This shows that stability has to do with the power boundedness property of the matrix $\mathbf{E}$ [Trefethen, 1996, Chapter 4], [LeVeque and Trefethen, 1984, Strikwerda and Wade, 1994, Kraaijevanger, 1994]. In our case, thanks to the Parseval equality, it reads

$$\sup_{t \in [\![1, n_T]\!]\Delta t} \sup_{\theta \in [-\pi, \pi]} |||\hat{\mathbf{E}}(\theta)^{t/\Delta t}|||_2 \leq C_T.$$

However, the submultiplicativity of the norm does not allow to conclude, due to (9.3), since

$$|||\hat{\mathbf{E}}(\theta)^{t/\Delta t}|||_2 \leq |||\hat{\mathbf{E}}(\theta)|||_2^{t/\Delta t} = 2^{t/(2\Delta t)},$$

for every $t \in [\![1, n_T]\!]\Delta t$, where the right hand side goes to infinity for small Δt. This confirms that one cannot deduce stability from the value of $|||\hat{\mathbf{E}}(\theta)|||_2$, and that is why the lattice Boltzmann community avoids doing so. The right notion to use is power boundedness, which is however difficult to check in general but linked to the spectrum of $\mathbf{E}$ rather than to its L^2 norm. In our case, it can be easily checked, since

$$\hat{\mathbf{E}}(\theta)^{t/\Delta t} = \begin{bmatrix} e^{-it/\Delta t\theta} & 0 \\ e^{-it/\Delta t\theta} & 0 \end{bmatrix}, \qquad \text{hence} \qquad |||\hat{\mathbf{E}}(\theta)^{t/\Delta t}|||_2 = \sqrt{2} = C_T = C,$$

and deducing that the scheme is stable. To our understanding, the misleading indications given by the norm $|||\hat{\mathbf{E}}(\theta)|||_2$ come from the fact that the lattice Boltzmann scheme enlarges the size of the state space from N to q, with the presence of intrinsically numerical—yet nonphysical—unknowns. Moreover, the matrices $\mathbf{E}$ are practically never symmetric when solving hyperbolic equations, creating a gap between spectral norms and spectral radii.

9.2 One conserved moment

Let us start by studying the case $N = 1$. We recall that—in this setting—we are going to show that the notion of *von Neumann* linear stability for a lattice Boltzmann scheme is totally equivalent to the one for its corresponding Finite Difference scheme.

9.2.1 Stability

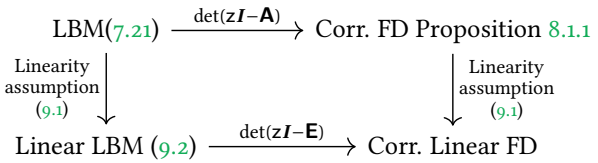
The fundamental result is the following.

Proposition 9.2.1: Amplification polynomial

Consider $N = 1$ and a linear or linearized lattice Boltzmann scheme under the form (9.2). The amplification polynomial of the corresponding Finite Difference scheme given by Proposition 8.1.1, denoted by $\hat{\Phi}$, see (7.41), equals the characteristic polynomial of $\hat{\mathbf{E}}$, that is

$$\hat{\Phi}(\boldsymbol{\theta}, z) = \det(z\boldsymbol{I} - \hat{\mathbf{E}}(\boldsymbol{\theta})).$$

Thus, any property on the spectrum of the scheme matrix $\hat{\mathbf{E}}$ for the lattice Boltzmann scheme shall find an exact counterpart in the roots of the amplification polynomial of its corresponding Finite Difference scheme. Observe that operating in the primal space—by considering $\mathbf{E}$—or in the Fourier space—taking $\hat{\mathbf{E}}$—is exactly the same thing. The meaning of Proposition 9.2.1 can be explained by the following diagram:



The corresponding Finite Difference scheme will be the same if we first compute it through Proposition 8.1.1 without linearity assumption and then we finally assume that the equilibria are linear (9.1) (right-down movement); or if we first rewrite the lattice Boltzmann with the scheme matrix $\mathbf{E}$ and (9.2), assuming from the very beginning the it is linear and then use its determinant to get rid of the non-conserved moments.

Proof of Proposition 9.2.1. By looking at the corresponding Finite Difference scheme in Proposition 8.1.1 given by (8.2), we obtain that

$$\Phi(z)m_1 = \det(z\boldsymbol{I}-\mathbf{A})m_1 - (\operatorname{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_1 = \det(z\boldsymbol{I}-\mathbf{A})m_1 - (\operatorname{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{B}\boldsymbol{\epsilon}_1\otimes\boldsymbol{e}_1)_{11}m_1,$$

where the second equality is obtained by assuming linearity of the equilibria (9.1). This entails

$$\Phi(z) = \det(z\boldsymbol{I}-\mathbf{A}) - \boldsymbol{e}_1^{\mathrm{t}}\operatorname{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{B}\boldsymbol{\epsilon}_1\otimes\boldsymbol{e}_1 = \det(z\boldsymbol{I}-\mathbf{A}-\boldsymbol{\epsilon}_1\otimes\boldsymbol{e}_1) = \det(z\boldsymbol{I}-\mathbf{E}),$$

using Lemma 8.2.1 to achieve the second equality. □

Since the Finite Difference scheme and the original lattice Boltzmann scheme share exactly the same spectrum, according to Proposition 9.2.1, any definition of stability based on it shall be perfectly the same for the two entities. Let us therefore discuss them.

If we want to make Definition 9.1.1 more accurate, in order to perfectly match Theorem 7.7.3 taken from [Strikwerda, 2004], we have:

Definition 9.2.1: Restricted *von Neumann* stability for lattice Boltzmann

Consider a linear or linearized lattice Boltzmann scheme, which therefore can be put under the form (9.2). We say that it is stable in the *von Neumann* sense (with restricted condition) if, for every $\boldsymbol{\theta}\in[-\pi,\pi]^d$, every $\hat{\mathrm{g}}_k(\boldsymbol{\theta})\in\operatorname{sp}(\hat{\mathbf{E}}(\boldsymbol{\theta}))$ for $k\in[\![1,q]\!]$, is such that:

1. $|\hat{\mathrm{g}}_k(\boldsymbol{\theta})|\leq 1$.
2. If $|\hat{\mathrm{g}}_k(\boldsymbol{\theta})|=1$, then $\hat{\mathrm{g}}(\boldsymbol{\theta})$ is a simple eigenvalue of $\hat{\mathbf{E}}(\boldsymbol{\theta})$.

Usually—within the lattice Boltzmann community—only the first condition in Definition 9.2.1 is actually checked, see [Février, 2014, Equation (1.57) and Remark 1.4.4] and Definition 9.1.1. We added the second condition in order to be more precise on the subtle question of multiple eigenvalues, by bringing this definition closer to Theorem 7.7.3. This subtlety arises when considering multi-step schemes. Still, this question is not harmless since for instance the $\mathrm{D_1Q_2}$ scheme rewrites as a leap-frog scheme if the relaxation parameter is equal to two. This very Finite Difference scheme can suffer from linear growth of the solution due to this issue, see [Strikwerda, 2004, Chapter 4]. What is very appealing is that Definition 9.2.1 offers, thanks to Theorem 7.7.3, necessary and sufficient conditions in the case $N=1$, when the scheme does not depend on the time and space steps, which shall not be the case when $N>1$.

We observe that when the corresponding Finite Difference scheme depends explicitly on Δt or Δx, one can still have a necessary and sufficient condition by taking this into account:

Theorem 9.2.1: *Von Neumann* stability

Let $N=1$. If the amplification polynomial $\hat{\Phi}(\boldsymbol{\theta},z)$ depends on Δt and Δx, then the necessary and sufficient conditions for stability of the multi-step scalar linear Finite Difference scheme—*cf.* Proposition 8.1.1—for the ℓ^2 norm is that all the roots $\hat{\mathrm{g}}_k(\boldsymbol{\theta})$ for $k\in[\![1,q]\!]$ of $\hat{\Phi}(\boldsymbol{\theta},z)$ satisfy the following conditions.

1. There is a constant $\alpha>0$ such that $|\hat{\mathrm{g}}_k(\boldsymbol{\theta})|\leq 1+\alpha\Delta t$.

2. There are constants $c_0, c_1 > 0$ such that if $c_0 \leq |\hat{\mathrm{g}}_k(\boldsymbol{\theta})| \leq 1 + \alpha\Delta t$, then $\hat{\mathrm{g}}_k(\boldsymbol{\theta})$ is a simple root, and for any other root $\hat{\mathrm{g}}_r(\boldsymbol{\theta})$, the relation
$$|\hat{\mathrm{g}}_k(\boldsymbol{\theta}) - \hat{\mathrm{g}}_r(\boldsymbol{\theta})| \geq c_1,$$
holds for Δt and Δx small enough.

This result is [Strikwerda, 2004, Theorem 4.2.2] and states that we still have stability if the eigenvalues are slightly larger than one in modulus but ...how larger: proportionally to Δt. Moreover, one needs to have simple roots which are sufficiently "spaced".

As previously discussed, Proposition 9.2.1 readily gives a Corollary, stating that in the case $N = 1$, the stability of the lattice Boltzmann scheme according to Definition 9.2.1 is equivalent to the *von Neumann* stability of its corresponding Finite Difference scheme—given by Proposition 8.1.1—according to the definition of stability given in Theorem 7.7.3.

Corollary 9.2.1: Equivalence between stabilities

Consider $N = 1$ and a linear lattice Boltzmann scheme under the form (9.2). The lattice Boltzmann scheme is stable in the *von Neumann* sense according to Definition 9.2.1 if and only if its corresponding Finite Difference scheme given by Proposition 8.1.1 is *von Neumann* stable according to Theorem 7.7.3. More schematically

Stable lattice Boltzmann scheme $\quad\rightleftarrows\quad$ Stable corresponding Finite Difference scheme.

In terms of spectrum
$$\{\text{roots of } \hat{\Phi}(\boldsymbol{\theta}, \mathsf{z})\} \equiv \mathrm{sp}(\hat{\mathbf{E}}(\boldsymbol{\theta})).$$

This confirms that the usual notion of stability for lattice Boltzmann is the right one and finds a direct analogue in the theory of Finite Difference schemes.

9.2.2 Uniqueness of the corresponding Finite Difference scheme

Something that we have left when we stated and proved Proposition 7.5.1 and Proposition 8.1.1 was whether the Finite Difference that they provide is unique in some sense. We now try to answer this question for $N = 1$.

The matrix-determinant Lemma 8.2.1 actually provides the way of transferring information—as far as (8.2) is concerned—from the term in the determinant (used to eliminate the non-conserved moments) and the term with the adjugate (treated as "slave" part). Consider an additive decomposition of $\mathbf{A}$ analogous to (7.29), under the form
$$\mathbf{A} = (\mathbf{A} - \mathbf{d} \otimes \boldsymbol{e}_1) + \mathbf{d} \otimes \boldsymbol{e}_1,$$
for whatever $\mathbf{d} \in \mathsf{D}^q$, where $\mathbf{A} - \mathbf{d} \otimes \boldsymbol{e}_1$ plays the role of $\mathbf{A}_i$ and $\mathbf{d} \otimes \boldsymbol{e}_1$ the one of $\mathbf{A}_i^\diamond$ in (7.29). In this way, we "save" the term $\mathbf{d} \otimes \boldsymbol{e}_1$, which is harmless, because it depends solely on the conserved moment that we eventually want to keep. One thus writes the lattice Boltzmann scheme as
$$(\mathsf{z}\boldsymbol{I} - \mathbf{A} + \mathbf{d} \otimes \boldsymbol{e}_1)\mathbf{m} = \mathbf{d}\mathsf{m}_1 + \mathbf{B}\mathbf{m}^{\mathrm{eq}}. \tag{9.5}$$
Multiplying by $\mathrm{adj}(\mathsf{z}\boldsymbol{I} - \mathbf{A} + \mathbf{d} \otimes \boldsymbol{e}_1)$, using (8.3) and selecting the first row provides, as usual
$$\det(\mathsf{z}\boldsymbol{I} - \mathbf{A} + \mathbf{d} \otimes \boldsymbol{e}_1)\mathsf{m}_1 = (\mathrm{adj}(\mathsf{z}\boldsymbol{I} - \mathbf{A} + \mathbf{d} \otimes \boldsymbol{e}_1)\mathbf{d})_1 \mathsf{m}_1 + (\mathrm{adj}(\mathsf{z}\boldsymbol{I} - \mathbf{A} + \mathbf{d} \otimes \boldsymbol{e}_1)\mathbf{B}\mathbf{m}^{\mathrm{eq}})_1. \tag{9.6}$$
At first sight, it seems that we have constructed another Finite Difference scheme (9.6) which is *a priori* different from (8.2). Let us show that the schemes are indeed equal. By construction of the adjugate matrix as the transpose of the cofactor matrix, perturbing the first column does not impact the first row of the adjugate, that is
$$\mathrm{adj}(\mathsf{z}\boldsymbol{I} - \mathbf{A} + \mathbf{d} \otimes \boldsymbol{e}_1)_{1,\cdot} = \mathrm{adj}(\mathsf{z}\boldsymbol{I} - \mathbf{A})_{1,\cdot},$$

providing

$$\begin{aligned}\det(z\boldsymbol{I}-\mathbf{A}+\mathbf{d}\otimes\boldsymbol{e}_1)\mathsf{m}_1 &= (\mathrm{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{d})_1\mathsf{m}_1+(\mathrm{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_1\\ &= \det(z\boldsymbol{I}-\mathbf{A}+\mathbf{d}\otimes\boldsymbol{e}_1)\mathsf{m}_1-\det(z\boldsymbol{I}-\mathbf{A})\mathsf{m}_1+(\mathrm{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_1,\end{aligned}$$

using the matrix-determinant Lemma 8.2.1 to pass from the first to the second line. We have then obtained

$$\det(z\boldsymbol{I}-\mathbf{A})\mathsf{m}_1=(\mathrm{adj}(z\boldsymbol{I}-\mathbf{A})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_1,$$

coinciding with (8.2), as claimed. This shows that in the scalar case $N=1$, no matter how we eliminate the non-conserved moments by "saving" some information concerning the conserved moment m_1 on the right hand side of (9.5), the corresponding Finite Difference scheme will always be the same. This is unfortunately false—as we shall demonstrate—when we deal with several conserved moments $N>1$. To sum up, for $N=1$, no matter when the linearity assumption (9.1) is done and which matrix is used to eliminate the non-conserved variables: the corresponding Finite Difference scheme shall always be the same.

9.3 Several conserved moments

We now consider $N>1$, namely there are several conserved moments in the scheme at hand. For $N>1$, the situation is different from $N=1$: there exists a plethora of ways of proposing corresponding Finite Difference schemes. Still, for our way of proceeding, *cf.* Proposition 8.1.2, we can show that the stability of the corresponding Finite Difference implies that of the lattice Boltzmann scheme.

9.3.1 *Von Neumann* stability for Finite Difference scheme for systems

In order to define a notion of ℓ^2 stability for multi-step Finite Difference scheme with several unknowns, we follow and adapt the construction presented in [Gustafsson et al., 1995, Chapter 5]. Analogously to (7.40), the Finite Difference scheme that we obtain by applying Proposition 8.1.2 and eventually assuming that the equilibria are linear, see (9.1), reads

$$\sum_{k=0}^{q-N+1}\boldsymbol{\varphi}_k z^k\begin{bmatrix}\mathsf{m}_1\\ \vdots\\ \mathsf{m}_N\end{bmatrix}=0, \tag{9.7}$$

where $(\boldsymbol{\varphi}_k)_{k\in[\![0,q-N+1]\!]}\subset\mathcal{M}_N(\mathsf{D})$ are matrices of spatial Finite Difference operators. Observe that since all our schemes are explicit, we have that $\boldsymbol{\varphi}_{q-N+1}=\boldsymbol{I}_N$. Having eliminated the non-conserved moments $\mathsf{m}_{N+1},\dots,\mathsf{m}_q$, in this Section 9.3, we indicate $\mathbf{m}=(\mathsf{m}_1,\dots,\mathsf{m}_N)^{\mathsf{t}}$, the vector of conserved moments which are kept in Proposition 8.1.2.

Example 9.3.1 ($\mathrm{D}_1\mathrm{Q}_3$ with $N=2$)**.** *Consider the corresponding Finite Difference scheme in Example 7.5.6. Take linear equilibria which allow to simulate the wave equation at velocity V under acoustic scaling, that is*

$$\boldsymbol{\epsilon}_1=\begin{bmatrix}1\\0\\V^2\end{bmatrix},\qquad \boldsymbol{\epsilon}_2=\begin{bmatrix}0\\1\\0\end{bmatrix}.$$

We obtain

$$\boldsymbol{\varphi}_2=\boldsymbol{I}_2,\ \boldsymbol{\varphi}_1=\begin{bmatrix}-1-(1-s_3)\mathsf{S}(\mathsf{x}_1)-\frac{s_3V^2}{\lambda^2}(\mathsf{S}(\mathsf{x}_1)-1) & -\frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1)\\ -\frac{s_3V^2}{\lambda}\mathsf{A}(\mathsf{x}_1) & -(2-s_3)\mathsf{S}(\mathsf{x}_1)\end{bmatrix},\ \boldsymbol{\varphi}_0=\begin{bmatrix}(1-s_3)\mathsf{S}(\mathsf{x}_1) & \frac{(1-s_3)}{\lambda}\mathsf{A}(\mathsf{x}_1)\\ 0 & 1-s_3\end{bmatrix}.$$

As customary with multi-step schemes, we consider an extended variable spanning the discrete solution on

several time steps. Following [Gustafsson et al., 1995, Equation (5.1.3)], we set

$$\tilde{\mathbf{m}}(t) := (\mathsf{z}^{q-N+1}\mathbf{m}, \cdots, \mathbf{m})^{\mathsf{t}}(t) = \begin{bmatrix} \mathsf{m}_1(t+(q-N+1)\Delta t) \\ \vdots \\ \mathsf{m}_N(t+(q-N+1)\Delta t) \\ \vdots \\ \vdots \\ \mathsf{m}_1(t) \\ \vdots \\ \mathsf{m}_N(t) \end{bmatrix},$$

thus we can recast (9.7) under a one-step form which reads

$$\mathsf{z}\tilde{\mathbf{m}} = \mathbf{Q}\tilde{\mathbf{m}}, \qquad \text{where} \qquad \mathbf{Q} = \begin{bmatrix} -\boldsymbol{\varphi}_{q-N} & \cdots & -\boldsymbol{\varphi}_2 & -\boldsymbol{\varphi}_1 & -\boldsymbol{\varphi}_0 \\ \boldsymbol{I}_N & \cdots & \mathbf{0}_N & \mathbf{0}_N & \mathbf{0}_N \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0}_N & \cdots & \boldsymbol{I}_N & \mathbf{0}_N & \mathbf{0}_N \\ \mathbf{0}_N & \cdots & \mathbf{0}_N & \boldsymbol{I}_N & \mathbf{0}_N \end{bmatrix} \in \mathcal{M}_{N(q-N+1)}(\mathsf{D}),$$

is a block companion matrix. In order to introduce a metric for $\tilde{\mathbf{m}}$, one construct a norm in time which is nothing but the 2-norm on the $q-N+1$ discrete times spanned inside $\tilde{\mathbf{m}}$, that is

$$\|\tilde{\mathbf{m}}(t)\| := \left(\sum_{k=0}^{q-N+1} \|\mathbf{m}(t+k\Delta t)\|_{\ell^2}^2 \right)^{1/2}. \tag{9.8}$$

As previously highlighted, the definition of stability has to do with power boundedness, like it was the case for (9.4) in the case $N=1$. This is expressed by [Gustafsson et al., 1995, Definition 5.1.1]:

Definition 9.3.1: L^2 stability

The schemes (9.7) are said to be stable if (at least for Δx small enough), there exist constants K and α such that

$$|||\mathbf{Q}^{t/\Delta t}||| \leq Ke^{\alpha t}, \tag{9.9}$$

for $t \in \Delta t\mathbb{N}$, regardless of the value of Δt, where the norm $\|\cdot\|$ is the one induced by (9.8).

Remark 9.3.1 (Finite time horizon or not)**.** *This seems slightly different from the definition of stability (9.4) which has been previously given [Strikwerda, 2004]. It reads: the schemes (9.7) are said to be stable if (at least for Δx small enough), for every final time $T>0$, there exists C_T such that*

$$|||\mathbf{Q}^{t/\Delta t}||| \leq C_T, \tag{9.10}$$

for every $t \in [\![1, n_T]\!]\Delta t$, regardless of the value of Δt. In (9.9) one allows any time $t \in \Delta t\mathbb{N}$ (no fixed final time T) but with the upper bound on the right hand side depending on t exponentially. In fine*, the definitions are equivalent even when we use the scheme to approximate equations with exponentially growing solutions such as $\partial_t u + \partial_x u = u$. For this kind of problem, allowing exponential growth like in (9.9) or the constant C_T in (9.10) to depend on the final time horizon T is needed because the solution of the target equation grows exponentially in time.*

Since the schemes have constant coefficients, one can consider the problem in the Fourier space and thanks to the Parseval identity, one can prove [Gustafsson et al., 1995, Theorem 5.2.1]:

Theorem 9.3.1: L^2 stability with Fourier

The schemes (9.7) are stable according to Definition 9.3.1 if and only if (at least for Δx small enough)

$$|||\hat{\mathbf{Q}}(\boldsymbol{\theta})^{t/\Delta t}|||_2 \leq Ke^{\alpha t},$$

for every $t \in \Delta t\mathbb{N}$, regardless of the value of Δt and for every wave-number $\boldsymbol{\theta} \in [-\pi,\pi]^d$.

Remark 9.3.2 (Finite time horizon or not). *As previously pointed out, we could also write the previous result as: the schemes (9.7) are stable according to Definition 9.3.1 if and only if (at least for Δx small enough), for any $T > 0$*

$$|||\hat{\mathbf{Q}}(\boldsymbol{\theta})^{t/\Delta t}|||_2 \leq C_T,$$

for every $t \in [\![1, n_T]\!]\Delta t$, regardless of the value of Δt and for every wave-number $\boldsymbol{\theta} \in [-\pi,\pi]^d$.

However, these conditions are often difficult to check when $N > 1$ and the number of steps $q - N + 1$ grows. Thus, we look for a necessary condition which is easier to check, which yields the so-called *von Neumann* condition, given by [Gustafsson et al., 1995, Theorem 5.2.2]

Theorem 9.3.2: *von Neumann* stability

A necessary condition for the schemes (9.7) to be stable according to Definition 9.3.1 is that that any of the $N(q-N+1)$ eigenvalues $\hat{\mathrm{g}}_k(\boldsymbol{\theta}) \in \mathrm{sp}(\hat{\mathbf{Q}}(\boldsymbol{\theta}))$ for $k \in [\![1, N(q-N+1)]\!]$ of $\hat{\mathbf{Q}}(\boldsymbol{\theta})$ satisfies (at least for Δx small enough)

$$|\hat{\mathrm{g}}_k(\boldsymbol{\theta})| \leq e^{\alpha\Delta t}, \qquad \boldsymbol{\theta} \in [-\pi,\pi]^d.$$

This is called *von Neumann* condition.

Remark 9.3.3 (Restricted conditions). *Observe that in the limit of small Δt*

$$|\hat{\mathrm{g}}_k(\boldsymbol{\theta})| \leq e^{\alpha\Delta t} \leq 1 + \alpha\Delta t,$$

which is thus the general Neumann condition, cf. *Theorem 9.2.1. When the scheme is independent of Δt, we recover the well-known restricted condition $|\hat{\mathrm{g}}_k(\boldsymbol{\theta})| \leq 1$,* cf. *Theorem 7.7.3.*

Very often, especially for schemes which are both multi-step and with several variables, we cannot do better than this. Cases where the *von Neumann* condition is also sufficient are illustrated in [Gustafsson et al., 1995, Chapter 5]. When Δt and Δx are fixed, the conditions to have power boundedness are those in Theorem 7.7.3, but they need to be checked uniformly in Δt and Δx in order to be sufficient conditions, whence the assumption at the beginning of Theorem 7.7.3, which could now read as: $\hat{\mathbf{Q}}(\boldsymbol{\theta})$ explicitly independent of Δt and Δx. However, as claimed in [Strikwerda, 2004, Chapter 7], for $N > 1$, there is no good equivalent of Theorem 7.7.3 providing both necessary and sufficient conditions.

Let us go on. When the schemes are genuinely multi-step as in our case, it is better to work on the original scheme instead than on the companion matrix $\hat{\mathbf{Q}}(\boldsymbol{\theta})$. This can be done by the following lemma, see [Gustafsson et al., 1995, Lemma 5.2.2]:

Lemma 9.3.1: Characteristic equation

Introduce the matricial amplification polynomial, analogous to (7.41) for the case $N = 1$:

$$\hat{\boldsymbol{\Phi}}(\boldsymbol{\theta}, \mathsf{z}) := \sum_{k=0}^{q-N+1} \hat{\boldsymbol{\varphi}}_k(\boldsymbol{\theta})\mathsf{z}^k \in (\mathcal{M}_N(\hat{\mathrm{D}}))[\mathsf{z}],$$

with coefficients given by (9.7). The eigenvalues of the companion matrix $\hat{\mathbf{Q}}(\boldsymbol{\theta})$ are the roots of the char-

acteristic equation

$$\det(\hat{\mathbf{\Phi}}(\boldsymbol{\theta}, z)) = 0, \tag{9.11}$$

i.e.

$$\{\text{roots of } \det(\hat{\mathbf{\Phi}}(\boldsymbol{\theta}, z))\} \equiv \mathrm{sp}(\hat{\mathbf{Q}}(\boldsymbol{\theta})).$$

9.3.2 Lack of uniqueness of the corresponding Finite Difference scheme

Before bridging with the stability of the original lattice Boltzmann scheme, we need to spend some time on discussing the following fact. When $N > 1$, we lack uniqueness in the corresponding schemes and even the stage at which we assume linearity during the process of elimination of the non-conserved moments matters. Quite the opposite, in the case $N = 1$, we have observed that supposing that the scheme is linear at the very beginning or doing it at the very end of the elimination of the non-conserved moments gives the same outcome, thanks to the matrix-determinant Lemma 8.2.1. For the same reason, the corresponding Finite Difference scheme is the same if one additively splits the matrix $\mathbf{A}$ to put part of the dependence on the conserved moment in a sort of equilibrium term. Unfortunately, both of these claims are false for $N > 1$.

As far as the instant when we assume that (9.1) holds is concerned, we observe the following fact. Since the equilibria depend in general on several conserved variables, assuming linearity from the very beginning does not yield the same result as Proposition 8.1.2. To see this, take Example 9.3.1, where the third equilibrium depends only on the first conserved variable.

If we do not assume from the very beginning that the scheme is linear, we have proposed different additive splittings for the matrix $\mathbf{A}$, see Section 7.5.2, which are essentially different ways of "saving" some piece of information on the conserved moments that shall be multiplied by the adjugate matrix. Even in Section 7.5.2, we were already aware of this lack of uniqueness. All the splittings that we have envisioned fall into the following class:

Definition 9.3.2: Admissible additive matrix splitting

For every conserved moment spanned by $i \in [\![1, N]\!]$, we say that the additive matrix splitting

$$\mathbf{A} = \mathbf{A}_i + \mathbf{A}_i^{\diamond},$$

is admissible if

$$(\mathrm{adj}(z\boldsymbol{I}_q - \mathbf{A}_i)\mathbf{A}_i^{\diamond})_{ij} = 0, \qquad \text{for} \qquad j \in [\![N+1, q]\!].$$

This means that with an admissible splitting, one has successfully eliminated the non-conserved moments, whence the fact of looking for $j \in [\![N+1, q]\!]$. Under any admissible splitting, the corresponding Finite Difference scheme reads

$$\det(z\boldsymbol{I}_q - \mathbf{A}_i)\mathsf{m}_i - (\mathrm{adj}(z\boldsymbol{I}_q - \mathbf{A}_i)\mathbf{A}_i^{\diamond}\mathbf{m})_i - (\mathrm{adj}(z\boldsymbol{I}_q - \mathbf{A}_i)\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i = 0, \tag{9.12}$$

for $i \in [\![1, N]\!]$. Hitherto, we have analyzed two splittings. We add a third one for the sake of simplifying the proofs to come. They are:

- The trivial (naive) splitting that we have considered in Example 7.5.6, where

$$\mathbf{A}_i = \mathbf{A}, \tag{9.13}$$

  for every $i \in [\![1, N]\!]$.

- The splitting

$$\mathbf{A}_i = \mathbf{A}_{\{i\} \cup [\![N+1, q]\!]}, \qquad \text{where thus} \qquad (\mathbf{A}_i)_{rp} \begin{cases} 0, & \text{if either } r \in [\![1, N]\!] \setminus \{i\} \text{ or } p \in [\![1, N]\!] \setminus \{i\}, \\ \mathsf{A}_{rp}, & \text{otherwise,} \end{cases} \tag{9.14}$$

for $i \in [\![1, N]\!]$, which is the one that we recommended in Proposition 8.1.2. It can be easily shown that this splitting is admissible.

- The splitting
$$\mathbf{A}_i = \mathbf{A}_{\mathrm{NC}}, \qquad \text{with} \qquad \mathbf{A}_{\mathrm{NC}} := [\underbrace{\mathbf{0}_{q\times 1}|\cdots|\mathbf{0}_{q\times 1}}_{N \text{ columns}}|\mathbf{A}_{\cdot,N+1}|\cdots|\mathbf{A}_{\cdot,q}], \tag{9.15}$$
for $i \in [\![1, N]\!]$. Here, "NC" stands for "non-conserved". This splitting does the same thing for every conserved moments and is (trivially) admissible. The idea of this splitting is to keep uniquely the non-conserved moments—that shall be eliminated—in the first part $\mathbf{A}_i$ of the scheme.

Example 9.3.2. *Consider again the* $\mathrm{D_1Q_3}$ *with* $N = 2$ *introduced in Example 9.3.1. We can obtain different corresponding Finite Difference schemes. In particular:*

- *If we assume linearity from the very beginning and we do not save anything to be multiplied by the adjugate term, we obtain two schemes for* m_1 *and* m_2 *which are identical. The difference between them shall only arise from the different initializations they undergo, which is however not the topic of Chapter 9. We obtain the characteristic equation (9.11) given by*
$$\begin{aligned}\det[\det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E})\boldsymbol{I}_2] &= \det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E})^2 \\ &= \Big(\mathsf{z}^3 + \Big(-1 - (1-s_3)\mathsf{S}(\mathsf{x}_1) + \frac{s_2 V^2}{\lambda}(\mathsf{S}(\mathsf{x}_1) - 1)\Big)\mathsf{z}^2 + \Big(1 - s_3 + (2-s_3)\mathsf{S}(\mathsf{x}_1) - \frac{s_3}{\lambda^2}(\mathsf{S}(\mathsf{x}_1) - 2)\Big)\mathsf{z} \\ &\qquad - (1-s_3)\Big)^2 = 0. \end{aligned} \tag{9.16}$$

- *If we use the naive way of obtaining the Finite Difference scheme presented in Section 7.1.2, without linearity assumption from the very beginning, which also boils down to utilizing the splitting (9.13), we have what follows. We have already seen that the result* a priori *depends on the choice of* s_1, s_2, cf. *Example 7.5.6. We consider* $s_1, s_2 = 0$ *to provide the result, which reads*
$$\begin{aligned}\det\begin{bmatrix} \det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E}) & 0 \\ (\mathrm{adj}(\mathsf{z}\boldsymbol{I}_3 - \mathbf{A})\mathbf{B}\boldsymbol{\epsilon}_1)_2 & \det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{A}) \end{bmatrix} &= \det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{A})\det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E}) \\ &= (\mathsf{z}-1)(\mathsf{z}^2 - (2-s_3)\mathsf{S}(\mathsf{x}_1)\mathsf{z} + (1-s_3))\det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E}) = 0, \end{aligned} \tag{9.17}$$
where the way of writing the first entry $\det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E})$ *comes from the matrix-determinant Lemma 8.2.1.*

- *Using the corresponding Finite Difference scheme according to Proposition 8.1.2*—i.e. *one uses the splitting (9.14)*—*the characteristic equation reads*
$$\begin{aligned}&\det\begin{bmatrix} \mathsf{z}^2 + \Big(-1 - (1-s_3)\mathsf{S}(\mathsf{x}_1) - \frac{s_3 V^2}{\lambda^2}(\mathsf{S}(\mathsf{x}_1) - 1)\Big)\mathsf{z} + (1-s_3) & -\frac{1}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{z} + \frac{1-s_3}{\lambda}\mathsf{A}(\mathsf{x}_1) \\ -\frac{s_3 V^2}{\lambda}\mathsf{A}(\mathsf{x}_1)\mathsf{z} & \mathsf{z}^2 - (2-s_3)\mathsf{S}(\mathsf{x}_1)\mathsf{z} + (1-s_3) \end{bmatrix} \\ &= \mathsf{z}^2(\mathsf{z} - (1-s_3)\mathsf{S}(\mathsf{x}_1))\det(\mathsf{z}\boldsymbol{I}_3 - \mathbf{E}) = 0. \end{aligned} \tag{9.18}$$

The non-trivial roots of the characteristic equations for the different approaches are compared in Figure 9.1. Two eigenvalues of (9.16) are the physical ones, whereas the remaining one is numerical and only influences the stability of the method. As made explicit by the explicit expressions of (9.17) and (9.18), they share all the eigenvalues of (9.16), plus other fictitiously created ones which still remain stable, see Figure 9.1.

We have introduced (9.15) because this splitting is independent of the moment indices $i \in [\![1, N]\!]$. However, we are going to show that the corresponding Finite Difference schemes obtained by this splitting are the same as those obtained by (9.14) used in Proposition 8.1.2. A first step in this direction is to control the adjugate term, as given in:

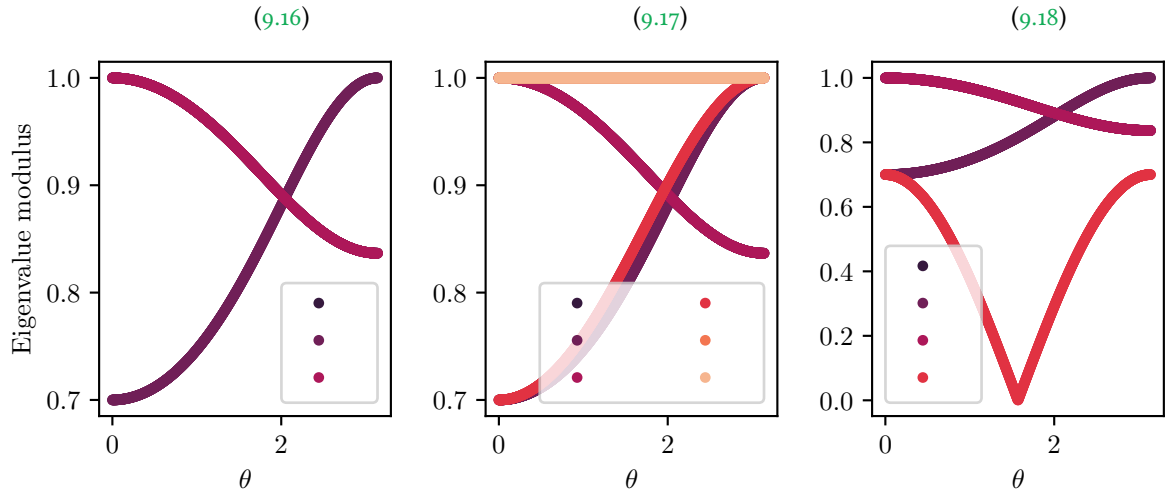


Figure 9.1: Modulus of the roots of different characteristic equations (9.16), (9.17) and (9.18) as function of the wavenumber $\theta \in [0, \pi]$ for some corresponding Finite Difference to the linear D_1Q_3 with $\lambda = 1$, $V = 1/2$ and $s_3 = 1.7$.

Lemma 9.3.2

Let $i \in [\![1, N]\!]$. The following identity holds:

$$\operatorname{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})_{i,\cdot} = \operatorname{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})_{i,\cdot}. \tag{9.19}$$

Proof. Let $i \in [\![1, N]\!]$. We have to compare the following adjugates

$$\operatorname{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]}) = \operatorname{adj}\left[\begin{array}{ccc|c|ccc||ccc} \mathsf{z} & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \mathsf{z} & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \hline 0 & \cdots & 0 & \mathsf{z}-\mathsf{A}_{ii} & 0 & \cdots & 0 & -\mathsf{A}_{i(N+1)} & \cdots & -\mathsf{A}_{iq} \\ \hline 0 & \cdots & 0 & 0 & \mathsf{z} & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & \mathsf{z} & 0 & \cdots & 0 \\ \hline\hline 0 & \cdots & 0 & -\mathsf{A}_{(N+1)i} & 0 & \cdots & 0 & \mathsf{z}-\mathsf{A}_{(N+1)(N+1)} & \cdots & -\mathsf{A}_{(N+1)q} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -\mathsf{A}_{qi} & 0 & \cdots & 0 & -\mathsf{A}_{q(N+1)} & \cdots & \mathsf{z}-\mathsf{A}_{qq} \end{array}\right],$$

and

$$\operatorname{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}}) = \operatorname{adj}\left[\begin{array}{ccc|c|ccc||ccc} \mathsf{z} & \cdots & 0 & 0 & 0 & \cdots & 0 & -\mathsf{A}_{1(N+1)} & \cdots & -\mathsf{A}_{1q} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \mathsf{z} & 0 & 0 & \cdots & 0 & -\mathsf{A}_{(i-1)(N+1)} & \cdots & -\mathsf{A}_{(i-1)q} \\ \hline 0 & \cdots & 0 & \mathsf{z} & 0 & \cdots & 0 & -\mathsf{A}_{i(N+1)} & \cdots & -\mathsf{A}_{iq} \\ \hline 0 & \cdots & 0 & 0 & \mathsf{z} & \cdots & 0 & -\mathsf{A}_{(i+1)(N+1)} & \cdots & -\mathsf{A}_{(i+1)q} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & \mathsf{z} & -\mathsf{A}_{N(N+1)} & \cdots & -\mathsf{A}_{Nq} \\ \hline\hline 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \mathsf{z}-\mathsf{A}_{(N+1)(N+1)} & \cdots & -\mathsf{A}_{(N+1)q} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & -\mathsf{A}_{q(N+1)} & \cdots & \mathsf{z}-\mathsf{A}_{qq} \end{array}\right].$$

We study each case

- We have, for the diagonal entry associated with the i-th moment:

$$\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})_{ii} = \det\left[\begin{array}{c|c} \mathsf{z}\boldsymbol{I}_{N-1} & \mathbf{0}_{(N-1)\times(q-N)} \\ \hline \mathbf{0}_{(q-N)\times(N-1)} & \mathbf{C}(\mathsf{z}) \end{array}\right],$$

$$\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})_{ii} = \det\left[\begin{array}{c|c} \mathsf{z}\boldsymbol{I}_{N-1} & \mathbf{D} \\ \hline \mathbf{0}_{(q-N)\times(N-1)} & \mathbf{C}(\mathsf{z}) \end{array}\right],$$

  hence the two quantities are equal, thanks to the formula for the determinant of block-triangular matrices.

- Let $j \in [\![1,N]\!] \setminus \{i\}$. By direct inspection, we see that $\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})_{ij} = 0$, since it comes from the determinant of a singular matrix which j-th column is zero. The analogous reason gives $\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})_{ij} = 0$, hence the thesis.

- Let $j \in [\![N+1,q]\!]$.

$$\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})_{ij} = \det\left[\begin{array}{c|c|c} \mathsf{z}\boldsymbol{I}_{i-1} & \mathbf{0}_{(i-1)\times(N-i)} & \mathbf{0}_{(i-1)\times(q-N)} \\ \hline \mathbf{0}_{1\times(i-1)} & \mathbf{0}_{1\times(N-i)} & -\mathbf{A}_{i,N+1\cdots q} \\ \hline \mathbf{0}_{(N-i)\times(i-1)} & \mathsf{z}\boldsymbol{I}_{N-i} & \mathbf{0}_{(N-i)\times(q-N)} \\ \hline \mathbf{0}_{(q-N-1)\times(i-1)} & \mathbf{0}_{(q-N-1)\times(N-i)} & \mathbf{C}(\mathsf{z}) \end{array}\right]$$

$$= \mathsf{z}^{i-1}\det\left[\begin{array}{c|c} \mathbf{0}_{1\times(N-i)} & -\mathbf{A}_{i,N+1\cdots q} \\ \hline \mathsf{z}\boldsymbol{I}_{N-i} & \mathbf{0}_{(N-i)\times(q-N)} \\ \hline \mathbf{0}_{(q-N-1)\times(N-i)} & \mathbf{C}(\mathsf{z}) \end{array}\right] = (-1)^{N-i}\mathsf{z}^{N-1}\det\left[\begin{array}{c} -\mathbf{A}_{i,N+1\cdots q} \\ \hline \mathbf{C}(\mathsf{z}) \end{array}\right],$$

  where $\mathbf{C}(\mathsf{z})$ depends on the choice of j and for the second equality, we have used the formula for the determinant of a block diagonal matrix and for the third one, we have performed several Laplace developments on the cofactor $(2,1)$.

$$\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})_{ij} = \det\left[\begin{array}{c|c|c} \mathsf{z}\boldsymbol{I}_{i-1} & \mathbf{0}_{(i-1)\times(N-i)} & -\mathbf{A}_{1\cdots i-1,N+1\cdots q} \\ \hline \mathbf{0}_{1\times(i-1)} & \mathbf{0}_{1\times(N-i)} & -\mathbf{A}_{i,N+1\cdots q} \\ \hline \mathbf{0}_{(N-i)\times(i-1)} & \mathsf{z}\boldsymbol{I}_{N-i} & -\mathbf{A}_{i+1\cdots N,N+1\cdots q} \\ \hline \mathbf{0}_{(q-N-1)\times(i-1)} & \mathbf{0}_{(q-N-1)\times(N-i)} & \mathbf{C}(\mathsf{z}) \end{array}\right]$$

$$= \mathsf{z}^{i-1}\det\left[\begin{array}{c|c} \mathbf{0}_{1\times(N-i)} & -\mathbf{A}_{i,N+1\cdots q} \\ \hline \mathsf{z}\boldsymbol{I}_{N-i} & -\mathbf{A}_{i+1\cdots N,N+1\cdots q} \\ \hline \mathbf{0}_{(q-N-1)\times(N-i)} & \mathbf{C}(\mathsf{z}) \end{array}\right] = (-1)^{N-i}\mathsf{z}^{N-1}\det\left[\begin{array}{c} -\mathbf{A}_{i,N+1\cdots q} \\ \hline \mathbf{C}(\mathsf{z}) \end{array}\right],$$

  with the same way of proceeding, thus yielding the claim.

This achieves the proof. □

With this result, we are done in showing that the corresponding Finite Difference scheme is the same for (9.14) and (9.15), as far as the equilibrium term $(\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)\mathbf{Bm}^{\mathrm{eq}})_i$ in (9.12) is concerned. Let us deal with the remaining one. The following result does not indeed assume any particular additive splitting for the matrix $\mathbf{A}$.

Lemma 9.3.3

Let $i \in [\![1,N]\!]$ and $\mathbf{A}_i$, $\mathbf{A}_i^\diamond$ fulfilling Definition 9.3.2. The following identity holds

$$\det(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)\boldsymbol{I}_q - \mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)\mathbf{A}_i^\diamond = \mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}). \tag{9.20}$$

Proof. Using the fundamental identity for the adjugate matrix (8.3), we obtain

$$\begin{aligned}\det(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)\boldsymbol{I}_q - \mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)\mathbf{A}_i^\diamond &= \det(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)\boldsymbol{I}_q - \mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)(\mathbf{A}_i^\diamond + \mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i - \mathsf{z}\boldsymbol{I}_q + \mathbf{A}_i) \\ &= -\mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)(\mathbf{A}_i^\diamond - \mathsf{z}\boldsymbol{I}_q + \mathbf{A}_i) = \mathrm{adj}(\mathsf{z}\boldsymbol{I}_q - \mathbf{A}_i)(\mathsf{z}\boldsymbol{I}_q + \mathbf{A}).\end{aligned}$$

□

All this together finally gives the claimed result.

Proposition 9.3.1: Equivalence of splitting (9.14) and (9.15)

The corresponding Finite Difference schemes (9.12) obtained for the splitting (9.14) and for (9.15) are the same, that is

$$\begin{aligned}\det(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathsf{m}_i \\ - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})(\mathbf{A} - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathbf{m})_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i \\ = \det(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\mathsf{m}_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathbf{A} - \mathbf{A}_{\mathrm{NC}})\mathbf{m})_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i,\end{aligned}$$

for every $i \in [\![1, N]\!]$.

Proof. We use all the previous results

$$\begin{aligned}\det(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathsf{m}_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})(\mathbf{A} - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathbf{m})_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i \\ \overset{(9.20)}{=} (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})(z\boldsymbol{I}_q - \mathbf{A})\mathbf{m})_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i \\ \overset{(9.19)}{=} (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(z\boldsymbol{I}_q - \mathbf{A})\mathbf{m})_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i \\ \overset{(9.20)}{=} \det(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\mathsf{m}_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathbf{A} - \mathbf{A}_{\{i\}\cup[\![N+1,q]\!]})\mathbf{m})_i - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\mathbf{B}\mathbf{m}^{\mathrm{eq}})_i.\end{aligned}$$

□

Hence, we can study the corresponding Finite Difference schemes given by Proposition 8.1.2 using the splitting (9.15). This is in general more handy to do.

9.3.3 Stability

We are now ready to bridge between lattice Boltzmann schemes and Finite Difference schemes when $N > 1$ in terms of stability. We form the characteristic equation (9.11) (in the primal space) for the schemes given by Proposition 8.1.2, which reads

$$\det(\boldsymbol{\Phi}(z)) := \det(\det(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\boldsymbol{I}_N - (\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathbf{E} - \mathbf{A}_{\mathrm{NC}}))_{1\cdots N,1\cdots N}) = 0, \tag{9.21}$$

thanks to Proposition 9.3.1.

Proposition 9.3.2: Spectral inclusion

The polynomial $\det(z\boldsymbol{I}_q - \mathbf{E})$ divides $\det(\boldsymbol{\Phi}(z))$ by (9.21). More explicitly

$$\det(\boldsymbol{\Phi}(z)) = \det(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})^{N-1}\det(z\boldsymbol{I}_q - \mathbf{E}).$$

Passing to the Fourier space, this means that all the zeros in $\det(z\boldsymbol{I}_q - \hat{\mathbf{E}}(\boldsymbol{\theta}))$ are also present in $\det(\hat{\boldsymbol{\Phi}}(\boldsymbol{\theta}, z))$. This is crucial to bridge between different definitions of stability.

Proof of Proposition 9.3.2. Using the fundamental identity of the adjugate (8.3), we have

$$\begin{aligned}\det(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\boldsymbol{I}_q - \operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathbf{E} - \mathbf{A}_{\mathrm{NC}}) &= \det(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\boldsymbol{I}_q - \operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}} - z\boldsymbol{I}_q + \mathbf{E}) \\ &= \operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(z\boldsymbol{I}_q - \mathbf{E}),\end{aligned}$$

hence

$$\det(\boldsymbol{\Phi}(z)) = \det((\operatorname{adj}(z\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(z\boldsymbol{I}_q - \mathbf{E}))_{1\cdots N,1\cdots N}).$$

We cannot directly use the formula for the determinant of a product since after the matrix multiplication in the previous expression, the resulting matrix is trimmed by taking the left upper sub-matrix of size N. Thanks to the structure of $\mathrm{adj}(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})$ and $\mathbf{E}$, we obtain

$$\mathrm{adj}(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathbf{E} - \mathbf{A}_{\mathrm{NC}}) = \left[\begin{array}{c|c} \mathbf{C} & \mathbf{0}_{N\times(q-N)} \\ \hline \mathbf{D} & \mathbf{0}_{(q-N)\times(q-N)} \end{array}\right],$$

hence

$$\mathrm{adj}(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathrm{z}\boldsymbol{I}_q - \mathbf{E}) = \left[\begin{array}{c|c} \tilde{\mathbf{C}} & \mathbf{0}_{N\times(q-N)} \\ \hline \mathbf{D} & \det(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})\boldsymbol{I}_{q-N} \end{array}\right].$$

Therefore, using [Horn and Johnson, 2012, Equation (0.8.5.8)] which gives an explicit expression for the determinant of an adjugate matrix as function of the determinant itself and the size of the matrix, we have:

$$\begin{aligned}\det(\mathrm{adj}(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})(\mathrm{z}\boldsymbol{I}_q - \mathbf{E})) = \det(\mathrm{adj}(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}}))\det(\mathrm{z}\boldsymbol{I}_q - \mathbf{E}) &= \det(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})^{q-1}\det(\mathrm{z}\boldsymbol{I}_q - \mathbf{E}) \\ &= \det(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})^{q-N}\det(\tilde{\mathbf{C}}).\end{aligned}$$

This entails

$$\det(\boldsymbol{\Phi}(\mathrm{z})) = \det(\tilde{\mathbf{C}}) = \det(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})^{N-1}\det(\mathrm{z}\boldsymbol{I}_q - \mathbf{E}),$$

yielding the divisibility property and an explicit expression for the divisor. □

Concerning the additional eigenvalues—observed in Example 9.3.2 and Figure 9.1 for a specific scheme—that are generated when considering the corresponding Finite Difference scheme *in lieu* of the original lattice Boltzmann scheme and contained in $\det(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})^{N-1}$, we have

$$\det(\mathrm{z}\boldsymbol{I}_q - \mathbf{A}_{\mathrm{NC}})^{N-1} = \mathrm{z}^{N(N-1)}\det(\mathrm{z}\boldsymbol{I}_{q-N} - \mathbf{A}_{N+1\cdots q,N+1\cdots q}) = \mathrm{z}^{N(N-1)}\det(\mathrm{z}\boldsymbol{I}_{q-N} - \mathbf{E}_{N+1\cdots q,N+1\cdots q}),$$

using the fact that the last $q-N$ columns of $\mathbf{B}\sum_{i=1}^{i=N}\boldsymbol{e}_i\otimes\boldsymbol{e}_i$ in (9.2) are zero. They are almost the eigenvalues of $\mathbf{E}$ but this matrix is trimmed, thus it is hard to explicitly characterize them. We could try to gain some control on these additional eigenvalues by showing that

$$\det(\mathrm{z}\boldsymbol{I}_{q-N} - \hat{\mathbf{E}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta})),$$

is a simple *von Neumann* polynomial, *cf.* Definition 7.7.1. If we would like to use Theorem 7.7.2, we have to compare with the polynomial

$$\mathrm{z}^{q-N}\overline{\det(\overline{\mathrm{z}}^{-1}\boldsymbol{I}_{q-N} - \hat{\mathbf{E}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))} = \det(\boldsymbol{I}_{q-N} - \mathrm{z}\hat{\mathbf{E}}^{*}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta})),$$

where we used the fact that the determinant is a multi-linear function. According to Theorem 7.7.2, we would enforce

$$\begin{aligned}|\det(\mathrm{z}\boldsymbol{I}_{q-N} - \hat{\mathbf{E}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))|_{\mathrm{z}=0} = |\det(\hat{\mathbf{E}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))| &= |\det(\hat{\mathbf{T}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))|\prod_{i=N+1}^{q}|1-s_i| \\ &< |\det(\boldsymbol{I}_{q-N} - \mathrm{z}\hat{\mathbf{E}}^{*}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))|_{\mathrm{z}=0} = |\det(\boldsymbol{I}_{q-N})| = 1.\end{aligned}$$

The issue is that, even if we easily have that (indeed $\hat{\mathbf{T}}$ is a unitary matrix)

$$|\det(\hat{\mathbf{T}}(\boldsymbol{\theta}))| = |\det(\boldsymbol{M})|\left|\prod_{j=1}^{q}\hat{\mathrm{t}}_{\boldsymbol{c}_j}(\boldsymbol{\theta})\right||\det(\boldsymbol{M}^{-1})| = 1,$$

we cannot straightforwardly conclude that $|\det(\hat{\mathbf{T}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))| = 1$ as well, even if this is true for Example 9.3.2. If we assume that this latter equality holds, a necessary condition (besides the second possible condition

in Theorem 7.7.2) to control the remaining eigenvalues is to have

$$\prod_{i=N+1}^{q} |1-s_i| < 1,$$

which tells us that selecting relaxation parameters in $]0,2[$ goes in the right direction of allowing to control the additional eigenvalues created by the fact of recasting the scheme as Finite Difference. We were however not capable of exploiting Theorem 7.7.2 any further. It is extremely difficult to deal with the eigenvalues of $\mathbf{A}$ in full generality because there is no simple relation for the eigenvalues of the product of a matrix and a diagonal matrix.

We can finish on the following link between different stabilities.

Corollary 9.3.1: Link between stabilities

Consider $N > 1$ and a linear lattice Boltzmann scheme under the form (9.2). If the corresponding Finite Difference schemes by Proposition 8.1.2 are stable (necessary condition) for the *von Neumann* condition by Theorem 9.3.2 with $\alpha = 0$, then the original lattice Boltzmann scheme is stable according to *von Neumann*, see Definition 9.1.1. More schematically

Stable corresponding Finite Difference schemes $\quad\rightarrow\quad$ Stable lattice Boltzmann scheme.

In terms of spectrum

$$\{\text{roots of } \det(\hat{\mathbf{\Phi}}(\boldsymbol{\theta},z))\} \equiv \mathrm{sp}(\hat{\mathbf{E}}(\boldsymbol{\theta})) \cup \{\text{roots of } \det(z\boldsymbol{I}_{q-N} - \hat{\mathbf{E}}_{N+1\cdots q,N+1\cdots q}(\boldsymbol{\theta}))\} \cup \{0\}.$$

9.4 Back to the link $\mathrm{D}_d\mathrm{Q}_{2W+1}$ two-relaxation-times schemes with magic parameters equal to 1/4 for $N > 1$

In the present Section 9.4, we no longer assume that the equilibria are linear, *i.e.* that (9.1) holds. In Section 7.6.3 and in particular in Example 7.6.2, we have shown how to handle any link $\mathrm{D}_d\mathrm{Q}_{2W+1}$ two-relaxation-times scheme with magic parameters equal to $1/4$ in the case $N = 1$ in order to obtain a corresponding Finite Difference scheme with only two steps. We now illustrate how to obtain the same in the case $N > 1$, which was not possible before since $\Psi_{\mathbf{A}}$ does not entirely cancel the matrix $\mathbf{A}$, see (7.34). This will show that the Finite Difference schemes obtained in [Ginzburg, 2009] for $N > 1$ fit in our theory.

In order to obtain this, we have to take advantage of the fact that the corresponding Finite Difference scheme for $N > 1$ is not unique, *cf.* Section 9.3.2, thus a wisely chosen splitting according to Definition 9.3.2 can make the difference between obtaining the desired two-steps schemes or generic $(q-N+1)$-steps schemes by Proposition 7.5.2. However, the fact that the moment matrix $\boldsymbol{M}$ that we have considered does not define the conserved moments—except the first one—*i.e.* holds only for $i = 1$, is a difficulty to address.

The idea orally suggested by I. Ginzburg is to take advantage of the fact that—since (1.3) holds for $i = 1$—we have $m_1^{\mathrm{eq}} = \mathsf{m}_1$, and we can take the fictitious relaxation parameter of the first conserved moment to be equal to the ones of the moments with odd indices (the symmetric ones), which are $s_{2r+1} = 2 - s$ for $r \in [\![1, W]\!]$. This boils down to consider the splitting

$$\mathbf{A}_i = \left[\begin{array}{c|cc|c|cc} s-1 & \frac{1-s}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & \frac{s-1}{\lambda^2}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2})-1) & \cdots & \frac{1-s}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \frac{s-1}{\lambda^2}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}})-1) \\ \hline 0 & (1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & \frac{s-1}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\ 0 & \lambda(1-s)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & (s-1)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & (1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \frac{s-1}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) \\ 0 & & & & \lambda(1-s)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & (s-1)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}}) \end{array}\right], \qquad \mathbf{A}_i^{\diamond} = (2-s)\boldsymbol{e}_1 \otimes \boldsymbol{e}_1, \tag{9.22}$$

for every $i \in [\![1, q]\!]$, according to Definition 9.3.2, and not only for every $i \in [\![1, N]\!]$, because the moments m_2,

$\dots \mathsf{m}_N$ generated by the matrix $\boldsymbol{M}$ that we have selected are not the conserved ones. This splitting of the matrix $\mathbf{A}$ is clearly admissible by the particular form of $\mathbf{A}_i^\diamond$. It is straightforward to observe that $\mathbf{A}_i^2 = (s-1)^2\boldsymbol{I}$ for every $i \in [\![1,q]\!]$, which means that we have found the minimal polynomial of the matrix $\mathbf{A}_i$ which is $\mu_{\mathbf{A}_i}(\mathsf{z}) = \mathsf{z}^2-(s-1)^2$. This polynomial does the job for every moment since it cancels the whole matrix $\mathbf{A}_i$, contrarily to $\Psi_{\mathbf{A}}$ which just cancels the first row of $\mathbf{A}$.

The corresponding Finite Difference schemes obtained using $\mu_{\mathbf{A}_i}$ are given by an expression similar to (7.31), where we utilize the coefficients of $\mu_{\mathbf{A}_i}$ instead of those of the characteristic polynomial of $\mathbf{A}_i$, where $\mathbf{A}_i$ and $\mathbf{A}_i^\diamond$ are given by (9.22), and $q-N$ is replaced by $\deg(\mu_{\mathbf{A}_i})-1=1$. This reads, again using $m_1^{\mathrm{eq}} = \mathsf{m}_1$

$$\begin{aligned}\mathsf{m}_i(t+\Delta t) = -\sum_{k=0}^{\deg(\mu_{\mathbf{A}_i})-1} \mathsf{q}_{i,k}\mathsf{m}_i(t+(1-\deg(\mu_{\mathbf{A}_i})+k)\Delta t)\\ +\left(\sum_{k=0}^{\deg(\mu_{\mathbf{A}_i})-1}\left(\sum_{r=0}^{k}\mathsf{q}_{i,\deg(\mu_{\mathbf{A}_i})+r-k}\mathbf{A}_i^r\right)(\mathbf{B}+\mathbf{A}_i^\diamond)\mathbf{m}^{\mathrm{eq}}(t-k\Delta t)\right)_i,\end{aligned}$$

for $i \in [\![1,q]\!]$, where $\mathsf{q}_{i,0} = -(s-1)^2$, $\mathsf{q}_{i,1}=0$, and $\mathsf{q}_{i,2}=1$. For $i=1$, after some computations, we obtain the scheme

$$\mathsf{m}_1(t+\Delta t) = (2-s)\mathsf{m}_1(t) + (s-1)\mathsf{m}_1(t-\Delta t) + \frac{s}{\lambda}\sum_{r=1}^{W}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r}^{\mathrm{eq}}(t) + \frac{2-s}{\lambda^2}\sum_{r=1}^{W}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})-1)\mathsf{m}_{2r+1}^{\mathrm{eq}}(t), \tag{9.23}$$

which unsurprisingly is (7.33), since this strategy encompasses the case $N=1$ and we have observed that uniqueness of the corresponding Finite Difference scheme—upon considering the same reduction strategy based either on the charactertistic polynomial or on the minimal polynomial—holds. The Finite Difference schemes for the other conserved moments are the new feature coming from the special additive splitting of $\mathbf{A}$ at hand. For $r \in [\![1,W]\!]$, analogous computations give the schemes for the antisymmetric moments for each block, which are:

$$\mathsf{m}_{2r}(t+\Delta t) = (s-1)^2\mathsf{m}_{2r}(t-\Delta t) + s\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r}^{\mathrm{eq}}(t) + \frac{2-s}{\lambda}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r+1}^{\mathrm{eq}}(t) + s(1-s)\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t). \tag{9.24}$$

For $r \in [\![1,W]\!]$, the schemes for the symmetric moments for each block are:

$$\mathsf{m}_{2r+1}(t+\Delta t) = (s-1)^2\mathsf{m}_{2r+1}(t-\Delta t) + \lambda s\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r}^{\mathrm{eq}}(t) + (2-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r+1}^{\mathrm{eq}}(t) + (s-1)(2-s)\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t).$$

These schemes are—using our notations—the same as (2.46) in [Ginzburg, 2009]. The number $N-1$ of conserved moments remaining after the first one is equal to d and these moments are not the antisymmetric moments m_{2r} for $r \in [\![1,W]\!]$ for each link, but rather the components of the momentum vector given by

$$\mathbf{q} := \sum_{j=1}^{q}\boldsymbol{\xi}_j\mathsf{f}_j = \sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{m}_{2r}.$$

Exceptionnally, as we have already pointed out, their conservation constraints are not written in the basis given by the moment matrix $\boldsymbol{M}$ and thus by (1.3), but rather by $\sum_{r=1}^{r=W}\boldsymbol{c}_{2r}\mathsf{m}_{2r}^{\mathrm{eq}} = \mathbf{q}$. Taking all this into account, the corresponding Finite Difference scheme for the remaining d conserved moments $\mathbf{q}$ (*i.e.* the momentum) is obtained by taking $\sum_{r=1}^{r=W}\boldsymbol{c}_{2r}\times$(9.24) and using the conservation constraints, yielding

$$\begin{aligned}\mathbf{q}(t+\Delta t) &= (s-1)^2\mathbf{q}(t-\Delta t) + s\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r}^{\mathrm{eq}}(t) + \frac{2-s}{\lambda}\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r+1}^{\mathrm{eq}}(t) + s(1-s)\overbrace{\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t)}^{=\mathbf{q}(t-\Delta t)}\\ &= (1-s)\mathbf{q}(t-\Delta t) + s\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r}^{\mathrm{eq}}(t) + \frac{2-s}{\lambda}\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\mathsf{m}_{2r+1}^{\mathrm{eq}}(t).\end{aligned} \tag{9.25}$$

The corresponding Finite Difference scheme (9.23) and (9.25) can also be recovered by a more "kinetic" standpoint by using the distribution functions f_j for $j \in [\![1,q]\!]$, which is arguably more handy than using moments. One step of the scheme written on the distribution functions reads

$$\begin{bmatrix} \mathsf{f}_1 \\ \mathsf{f}_2 \\ \mathsf{f}_3 \\ \vdots \\ \mathsf{f}_{2W} \\ \mathsf{f}_{2W+1} \end{bmatrix}(t+\Delta t) = \overbrace{\left[\begin{array}{c|cc|c|cc} s-1 & 0 & 0 & \cdots & 0 & 0 \\ \hline 0 & 0 & (s-1)\mathsf{t}_{\boldsymbol{c}_2} & & & \\ 0 & (s-1)\mathsf{t}_{-\boldsymbol{c}_2} & 0 & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & 0 & (s-1)\mathsf{t}_{\boldsymbol{c}_{2W}} \\ 0 & & & & (s-1)\mathsf{t}_{-\boldsymbol{c}_{2W}} & 0 \end{array}\right]}^{=\tilde{\mathbf{A}}} \begin{bmatrix} \mathsf{f}_1 \\ \mathsf{f}_2 \\ \mathsf{f}_3 \\ \vdots \\ \mathsf{f}_{2W} \\ \mathsf{f}_{2W+1} \end{bmatrix}(t) + \left[\begin{array}{c|cc|c|cc} 2-s & 0 & 0 & \cdots & 0 & 0 \\ \hline 0 & \mathsf{t}_{\boldsymbol{c}_2} & (1-s)\mathsf{t}_{\boldsymbol{c}_2} & & & \\ 0 & (1-s)\mathsf{t}_{-\boldsymbol{c}_2} & \mathsf{t}_{-\boldsymbol{c}_2} & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & \mathsf{t}_{\boldsymbol{c}_{2W}} & (1-s)\mathsf{t}_{\boldsymbol{c}_{2W}} \\ 0 & & & & (1-s)\mathsf{t}_{-\boldsymbol{c}_{2W}} & \mathsf{t}_{-\boldsymbol{c}_{2W}} \end{array}\right] \begin{bmatrix} \mathsf{f}_1^{\mathrm{eq}} \\ \mathsf{f}_2^{\mathrm{eq}} \\ \mathsf{f}_3^{\mathrm{eq}} \\ \vdots \\ \mathsf{f}_{2W}^{\mathrm{eq}} \\ \mathsf{f}_{2W+1}^{\mathrm{eq}} \end{bmatrix}(t).$$

Applying the scheme one more time gives:

$$\mathbf{f}(t+\Delta t) = (s-1)^2\mathbf{f}(t-\Delta t) + \left[\begin{array}{c|cc|c|cc} 2-s & 0 & 0 & \cdots & 0 & 0 \\ \hline 0 & \mathsf{t}_{\boldsymbol{c}_2} & (1-s)\mathsf{t}_{\boldsymbol{c}_2} & & & \\ 0 & (1-s)\mathsf{t}_{-\boldsymbol{c}_2} & \mathsf{t}_{-\boldsymbol{c}_2} & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & \mathsf{t}_{\boldsymbol{c}_{2W}} & (1-s)\mathsf{t}_{\boldsymbol{c}_{2W}} \\ 0 & & & & (1-s)\mathsf{t}_{-\boldsymbol{c}_{2W}} & \mathsf{t}_{-\boldsymbol{c}_{2W}} \end{array}\right] \mathbf{f}^{\mathrm{eq}}(t) \qquad (9.26)$$

$$+ \left[\begin{array}{c|cc|c|cc} (s-1)(2-s) & 0 & 0 & \cdots & 0 & 0 \\ \hline 0 & -(s-1)^2 & s-1 & & & \\ 0 & s-1 & -(s-1)^2 & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & -(s-1)^2 & s-1 \\ 0 & & & & s-1 & -(s-1)^2 \end{array}\right] \mathbf{f}^{\mathrm{eq}}(t-\Delta t).$$

Let us observe two facts. The first is that we directly see that the dependency on $\mathbf{f}$ has been diagonalized by going two time steps in the past: the distribution function of each population depends only on itself two steps before plus some terms depending on the conserved quantities. This is the analogous of having found the minimal polynomial $\mu_{\mathbf{A}_i}(\mathsf{z}) = \mathsf{z}^2 - (s-1)^2$ of $\mathbf{A}_i$, because the minimal polynomial is invariant under change of basis and one can easily check that $\tilde{\mathbf{A}} = \boldsymbol{M}^{-1}\mathbf{A}_i\boldsymbol{M}$ for every $i \in [\![1, q]\!]$. The second fact is that any linear combination of the rows of (9.26) gives a sort of Finite Difference scheme on this combination, thus in particular when this yields a conserved moment. Before writing the corresponding Finite Difference scheme on the conserved moments, we recall that

$$\boldsymbol{M}^{-1} = \left[\begin{array}{c|cc|c|cc} 1 & 0 & -\frac{1}{\lambda^2} & \cdots & 0 & -\frac{1}{\lambda^2} \\ \hline 0 & \frac{1}{2\lambda} & \frac{1}{2\lambda^2} & & & \\ 0 & -\frac{1}{2\lambda} & \frac{1}{2\lambda^2} & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & \frac{1}{2\lambda} & \frac{1}{2\lambda^2} \\ 0 & & & & -\frac{1}{2\lambda} & \frac{1}{2\lambda^2} \end{array}\right].$$

Considering that $\mathsf{m}_1 = \sum_{j=1}^{j=q} \mathsf{f}_j$ and using the fact that $m_1^{\mathrm{eq}} = \mathsf{m}_1$, we obtain that

$$\mathsf{m}_1(t+\Delta t) = (s-1)^2\mathsf{m}_1(t-\Delta t) + (2-s)\Big(\mathsf{m}_1(t) - \frac{1}{\lambda^2}\sum_{r=1}^{W} \mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big)$$

$$
\begin{aligned}
&+\sum_{r=1}^{W}\mathsf{t}_{\boldsymbol{c}_{2r}}\Big(\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)+(1-s)\Big(-\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big)\Big)\\
&+\sum_{r=1}^{W}\mathsf{t}_{-\boldsymbol{c}_{2r}}\Big(-\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)+(1-s)\Big(\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big)\Big)\\
&+(s-1)(2-s)\Big(\mathsf{m}_1(t-\Delta t)-\frac{1}{\lambda^2}\sum_{r=1}^{W}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t)\Big)-\frac{1}{\lambda^2}(s-1)^2\sum_{r=1}^{W}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t)+\frac{1}{\lambda^2}(s-1)\sum_{r=1}^{W}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t).
\end{aligned}
$$

After simplifying some terms and using the symmetric and antisymmetric parts of the shift operators associated with each pair of discrete velocities, we obtain (9.23) as expected. For the other conserved moments, taking $\mathbf{q} = \lambda\sum_{j=1}^{j=q}\boldsymbol{c}_j\mathsf{f}_j$, we obtain

$$
\begin{aligned}
\mathbf{q}(t+\Delta t) &= (s-1)^2\mathbf{q}(t-\Delta t)\\
&\quad+\lambda\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{t}_{\boldsymbol{c}_{2r}}\Big(\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)+(1-s)\Big(-\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big)\Big)\\
&\quad-\lambda\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{t}_{-\boldsymbol{c}_{2r}}\Big((1-s)\Big(\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big)-\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big)\\
&\quad+\lambda\sum_{r=1}^{W}\boldsymbol{c}_{2r}\Big(-(s-1)^2\Big(\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t)\Big)+(s-1)\Big(-\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t)\Big)\Big)\\
&\quad+\lambda\sum_{r=1}^{W}\boldsymbol{c}_{2r}\Big(-(s-1)\Big(\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t)\Big)+(s-1)^2\Big(-\frac{1}{2\lambda}\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t)+\frac{1}{2\lambda^2}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t-\Delta t)\Big)\Big).
\end{aligned}
$$

Simplifications yield

$$
\begin{aligned}
\mathbf{q}(t+\Delta t) &= (s-1)^2\mathbf{q}(t-\Delta t)+(s-s^2)\overbrace{\sum_{r=1}^{W}\boldsymbol{c}_{2r}\mathsf{m}_{2r}^{\mathrm{eq}}(t-\Delta t)}^{=\mathbf{q}(t-\Delta t)}\\
&\quad+\lambda\sum_{r=1}^{W}\boldsymbol{c}_{2r}\Big(\frac{s}{2}\mathsf{t}_{\boldsymbol{c}_{2r}}\mathsf{m}_{2r}^{\mathrm{eq}}(t)+\frac{2-s}{2\lambda}\mathsf{t}_{\boldsymbol{c}_{2r}}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)+\frac{s}{2}\mathsf{t}_{-\boldsymbol{c}_{2r}}\mathsf{m}_{2r}^{\mathrm{eq}}(t)-\frac{2-s}{2\lambda}\mathsf{t}_{-\boldsymbol{c}_{2r}}\mathsf{m}_{2r+1}^{\mathrm{eq}}(t)\Big),
\end{aligned}
$$

which becomes (9.25) as expected.

9.5 Conclusions of Chapter 9

In Chapter 9, we have shown that in the case of one conserved moment $N = 1$, the transformation to the corresponding Finite Difference scheme is unique and that the *von Neumann* stability of the original lattice Boltzmann scheme is totally equivalent to the *von Neumann* stability of the corresponding Finite Difference scheme. For several conserved moments $N > 1$, the corresponding Finite Difference scheme is no longer unique. For the one that we have proposed in Section 7.5—*cf.* Proposition 7.5.2—we show that the *von Neumann* stability of the corresponding Finite Difference scheme (which gives only necessary conditions in this context) implies the *von Neumann* stability of the original lattice Boltzmann scheme. The lack of uniqueness in the case $N > 1$ gives enough latitude to propose a different approach that encompasses the corresponding Finite Difference schemes found in [Ginzburg, 2009].

A future perspective of work is to use the corresponding Finite Difference scheme to analyze the stability with respect to other norms than the L^2 and to clarify the link with the weighted stability that has been discussed in the state of the art at the beginning of Chapter 9.

Chapter 10

Initialisation

General context and motivation

Numerical analysis features two notable frameworks where the knowledge of the initial state for numerical schemes is incomplete: one-step extended state-space methods (*e.g.* kinetic schemes, gas-kinetic schemes, *etc.*) and multi-step methods. On the one hand, lattice Boltzmann schemes have historically been considered in the realm of the one-step extended state-space methods [Kuznik et al., 2013]. From this standpoint, they have previously been compared [Graille, 2014, Simonis et al., 2020] to approximations of systems of conservation laws taking the form of relaxation systems *à la* Jin-Xin [Jin and Xin, 1995] and interpreted as peculiar discretisations of these systems when collision and transport terms are split and the relaxation time tends to zero proportionally to the time step. Both in the relaxation systems and the lattice Boltzmann schemes, conserved and non-conserved quantities are present at the same time but only conserved ones appear in the original system of conservation laws at hand. Although the initialisation of the non-conserved quantities remains free in principle, it has important repercussions on the behaviour of the solution—such as the formation of time boundary layers—both for the relaxation systems and the lattice Boltzmann schemes. On the other hand, in Chapter 7, lattice Boltzmann schemes have been thought and recast—as far as the evolution of the conserved quantities of interest is concerned—as multi-step Finite Difference schemes. Unsurprisingly, multi-step schemes both for Ordinary [Hundsdorfer and Ruuth, 2006, Hundsdorfer et al., 2003] and Partial Differential Equations [Gustafsson et al., 1995, Strikwerda, 2004] need to be properly initialised by some starting procedure with desired features, for example, consistency. When lattice Boltzmann schemes are seen in their original formulation, where conserved and non-conserved moments mingle, the initialisation of the non-conserved moments can be freely devised. Once the lattice Boltzmann schemes are recast as corresponding multi-step Finite Difference schemes solely on the conserved moments, *cf.* Chapter 7, the choice of initialisation for the conserved and non-conserved moments determines what the initialisation schemes feeding the corresponding bulk Finite Difference scheme at the beginning of the simulation are.

The previous discussion highlights that for numerical methods such as lattice Boltzmann schemes, the information gap between initial conditions for the target system of N conservation laws and the numerical method featuring q variables must be filled and thus the issue of providing decision tools to this end clearly manifests. Furthermore, one must be careful when comparing numerical schemes to the continuous problem they aim at approximating, because of the "more complicated physics" than the equations they are meant to simulate. The method of the modified equation is a valuable tool to describe this gap between numerical schemes and continuous equations and shall therefore be used to investigate the role of the initial conditions.

State of the art

In the framework of lattice Boltzmann schemes, previous efforts [Van Leemput et al., 2009] (under acoustic scaling), [Caiazzo, 2005, Junk and Yang, 2015, Huang et al., 2015b] (under diffusive scaling) have provided the first guidelines to establish the initial conditions, relying on asymptotic expansions both on the conserved and non-conserved variables. One aim of these studies has been to suppress initial oscillating boundary layers being part

of the "more complicated physics" of the discrete numerical method evoked in [Trefethen, 1996], which are however absent in the solution of the target conservation law. Moreover, since lattice Boltzmann schemes (respectively, their corresponding Finite Difference schemes) feature non-physical moments (respectively, parasitic modes/eigenvalues), these terms play a role in the consistency of the initialisation routines—contrarily to what happens in the bulk—creating a rich yet complex dynamics. Even if the techniques introduced in these works guarantee the elimination of the initial oscillating boundary phenomena, no precise quantitative analysis of their inner structure has been presented. Moreover, since the non-conserved moments do not have an analogue in the continuous problem, these procedures are—despite the fact of providing good indications—intrinsically formal. Finally, these works have only addressed the initialisation of specific lattice Boltzmann schemes, namely the D_1Q_2 for [Van Leemput et al., 2009], the D_1Q_2 and D_1Q_3 for [Junk and Yang, 2015] and the D_2Q_9 for [Caiazzo, 2005, Huang et al., 2015b].

Aim and structure of Chapter 10

Inspired by the open questions left by previous works in the literature, Chapter 10 aims at proposing a first general study on the initialisation of lattice Boltzmann schemes. The pivotal tool that we introduce is a modified equation analysis for the initial conditions/starting schemes and provides explicit constraints for general lattice Boltzmann schemes guaranteeing a sufficient order of consistency of the initialisation schemes to avoid order reduction of the overall method. The modified equations are obtained by considering that the choice of initial data shapes the starting schemes on the conserved variables of interest. Since the non-conserved moments are eliminated, the analyses we perform rely on less formal assumptions than the ones available in the literature. Pushing this tool one order further in the discretisation parameter, we meticulously describe the internal structure of the initial oscillating boundary layers, caused by incompatible numerical features—in particular, dissipation—between initialisation and bulk schemes. Previous works [Van Leemput et al., 2009] have certified the existence of these oscillations in numerical simulations without a thorough study of their structure. Let us insist once again on the fact that the dissipation of the physical mode for the initialisation schemes is driven both by the physical and parasitic eigenvalues of the bulk Finite Difference scheme. Another novelty in our work is the characterisation—by seeing lattice Boltzmann methods as dynamical systems on a commutative ring and exploiting the concept of observability—of a vast well-known class of lattice Boltzmann schemes (that has been already identified in Section 7.6.3) with a reduced number of initialisation schemes, irrespective of the number of non-conserved moments. The initial motivation to introduce the concept of observability is—for this class of schemes—to successfully determine the constraints needed to eliminate initial oscillating boundary layers due to the dissipation mismatch.

Chapter 10 is structured as follows. Section 10.1 fixes the target continuous problem of interest and recalls the basic needed elements concerning lattice Boltzmann scheme and corresponding Finite Difference schemes. This last point characterises the number of needed initialisation schemes. In Section 10.2, we introduce the modified equation analysis of these starting schemes and find the constraints under which they are consistent with the same equation as the bulk Finite Difference scheme. The examples and numerical simulations of Section 10.3 are introduced to corroborate the theoretical findings of Section 10.2 and—pushing the computation of the modified equations of the starting schemes one order further—we describe the internal structure of the initial oscillating boundary layers. One particular scheme stimulates the discussion of the following Section 10.4, where we re-evaluate the number of initialisation schemes at the discrete level more closely, thanks to the introduction of the notion of observability for the lattice Boltzmann schemes. This allows us to clearly identify and study a category of schemes for which the study of the initial conditions is greatly simplified and thus the constraints to avoid initial oscillating boundary layers can be easily established. We conclude in Section 10.5.

Contents

10.1 Target problem, lattice Boltzmann and corresponding Finite Difference schemes

10.1.1 Target problem

We consider lattice Boltzmann schemes with one conserved moment, for the sake of keeping the discussion and the notations simple and essential. The extension to several conserved moments can be envisioned in the spirit of Chapter 7 and Chapter 8. However, one has to be conscious of the specificity of $N > 1$, described in Chapter 9. We particularly concentrate on the widely adopted acoustic scaling [Dubois, 2022] between time and space steps. The diffusive scaling [Zhao and Yong, 2017, Zhang et al., 2019] is succinctly discussed with the very same techniques at the end of Section 10.4. Moreover, we consider linear schemes [Van Leemput et al., 2009], hence the equilibria for the non-conserved moments are linear functions of the conserved one, see (9.1). The lattice Boltzmann schemes we focus on aim at approximating the solution of the following linear Cauchy problem

$$\begin{cases} \partial_t u(t,\boldsymbol{x}) + \boldsymbol{V}\cdot\nabla_{\boldsymbol{x}} u(t,\boldsymbol{x}) = 0, & (t,\boldsymbol{x}) \in \mathbb{R}_+ \times \mathbb{R}^d, \quad (10.1) \\ u(0,\boldsymbol{x}) = u^\circ(\boldsymbol{x}), & \boldsymbol{x} \in \mathbb{R}^d, \quad (10.2) \end{cases}$$

with velocity $\boldsymbol{V} \in \mathbb{R}^d$ and initial datum u° which is a smooth function defined everywhere in $\mathbb{R}^d$. In this work, we only consider, contrarily to [Van Leemput et al., 2009], explicit initialisations, to keep the presentation simple. However, the analysis of implicit initialisations can be done with the same techniques.

10.1.2 Lattice Boltzmann schemes

We consider the schemes introduced in Chapter 1 with one conserved moment $N = 1$ and linear equilibria (9.1). For the sake of notations, we shall indicate the only equilibrium vector $\boldsymbol{\epsilon}_1$ simply by $\boldsymbol{\epsilon}$, since no ambiguity is possible. Therefore

$$\boldsymbol{m}^{\mathrm{eq}}(\mathsf{m}_1) = \boldsymbol{\epsilon} \otimes \boldsymbol{e}_1 \mathbf{m},$$

with $\epsilon_1 = 1$, *cf.* (1.3).

Algorithm 6 Lattice Boltzmann scheme.

- Given $\mathbf{m}(0,\boldsymbol{x}) \in \mathbb{R}^q$ for every $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$.
- For $n \in \mathbb{N}$
  - **Collision**. Using the collision matrix $\boldsymbol{K} := \boldsymbol{I} - \boldsymbol{S}(\boldsymbol{I} - \boldsymbol{\epsilon} \otimes \boldsymbol{e}_1)$ (*cf.* [Dubois and Lallemand, 2009, Equation (20)]), it reads
  $$\mathbf{m}^\star(n\Delta t,\boldsymbol{x}) = \boldsymbol{K}\mathbf{m}(n\Delta t,\boldsymbol{x}), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d. \tag{10.3}$$
  The post-collision distribution densities are recovered by $\mathbf{f}^\star(n\Delta t,\boldsymbol{x}) = \boldsymbol{M}^{-1}\mathbf{m}^\star(n\Delta t,\boldsymbol{x})$ on every point $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$ of the lattice.
  - **Transport**, which reads
  $$\mathsf{f}_j((n+1)\Delta t,\boldsymbol{x}) = \mathsf{f}_j^\star(n\Delta t,\boldsymbol{x} - \Delta x\boldsymbol{c}_j), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d, \quad j \in [\![1,q]\!]. \tag{10.4}$$
  The moments at the new time step are obtained by $\mathbf{m}((n+1)\Delta t,\boldsymbol{x}) = \boldsymbol{M}\mathbf{f}((n+1)\Delta t,\boldsymbol{x})$ on every point $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$.

The lattice Boltzmann scheme then reads as in Algorithm 6. For future use—analogously to Section 7.6.2—we introduce the number Q of non-conserved moments which do not relax to their equilibrium value during the collision phase (10.3):

$$Q := \#\{s_i \neq 1 \ : \ i \in [\![2,q]\!]\} \in [\![0,q[\![. \tag{10.5}$$

Roughly speaking, the larger Q, the stronger the "entanglement" between moments in the scheme. Remark that, since the corresponding column in $\boldsymbol{K}$ is zero, there is even no need to specify the initial value $\mathsf{m}_i(0,\cdot)$ when $s_i = 1$, for $i \in [\![2,q]\!]$. This comes from the fact that the post-collisional value of these moments is entirely determined by their value at equilibrium.

10.1.3 Corresponding Finite Difference scheme in the bulk

Using Proposition 7.5.1 and the observations presented in Section 9.2 concerning the uniqueness of the corresponding Finite Difference scheme in the scalar case, we have that the discrete dynamics of the conserved moment m_1 computed by Algorithm 6—away from the initial time—is the one of the corresponding Finite Difference scheme under the form

$$\mathsf{z}^{Q+1-q}\det(\mathsf{z}\boldsymbol{I} - \mathbf{E})\mathsf{m}_1(t,\boldsymbol{x}) = \sum_{k=0}^{q} \varphi_k \mathsf{z}^{k+Q+1-q}\mathsf{m}_1(t,\boldsymbol{x}) = 0, \qquad (t,\boldsymbol{x}) \in \Delta t\mathbb{N} \times \Delta x\mathbb{Z}^d, \tag{10.6}$$

where $(\varphi_k)_{k\in[\![0,q]\!]} \subset \mathsf{D}$ are the coefficients of the characteristic polynomial $\det(\mathsf{z}\boldsymbol{I} - \mathbf{E}) = \sum_{k=0}^{k=q} \varphi_k \mathsf{z}^k$ of $\mathbf{E}$, which is also the amplification polynomial (7.41). One can easily see that $\varphi_k = 0$ for $k \in [\![0, q-Q-2]\!]$, whence the important role played by Q. Furthermore, since the characteristic polynomial is monic, *i.e.* $\varphi_q = 1$, the scheme is explicit, thus can be recast into

$$\mathsf{z}\mathsf{m}_1(t,\boldsymbol{x}) = -\sum_{k=q-Q-1}^{q-1} \varphi_k \mathsf{z}^{k+1-q}\mathsf{m}_1(t,\boldsymbol{x}), \qquad (t,\boldsymbol{x}) \in \Delta t[\![Q,+\infty[\![\times \Delta x\mathbb{Z}^d.$$

We call this scheme corresponding bulk Finite Difference scheme acting on the bulk time steps $[\![Q,+\infty[\![$, which is a multi-step scheme with $Q+2$ stages. We remark the need for initialisation data through Q initialisation schemes, that we shall analyze in what follows.

10.1.4 Initialisation schemes

The initialisation schemes—the outcome of which eventually "nourishes" the bulk Finite Difference scheme—are determined by the choice of initial datum $\mathbf{m}(0,\cdot)$. They are:

$$\mathsf{m}_1(n\Delta t,\boldsymbol{x}) = (\mathbf{E}^n\mathbf{m})_1(0,\boldsymbol{x}), \qquad n \in [\![1,Q]\!], \quad \boldsymbol{x} \in \Delta x\mathbb{Z}^d.$$

The formulation we have proposed is provided in an abstract yet general form. In order to make the link with well-known lattice Boltzmann schemes and illustrate our purpose, let us introduce the following example. More of them are provided in Section 10.3 and Section 10.4.

Example 10.1.1 (D_1Q_2). *Consider the D_1Q_2 that we already analyzed in Example 7.5.1 with moment matrix taken in its dimensionless form. This gives*

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \quad \textit{Therefore} \quad \mathbf{T} = \begin{bmatrix} \mathsf{S}(\mathsf{x}_1) & \mathsf{A}(\mathsf{x}_1) \\ \mathsf{A}(\mathsf{x}_1) & \mathsf{S}(\mathsf{x}_1) \end{bmatrix}, \quad \textit{and} \quad \boldsymbol{K} = \begin{bmatrix} 1 & 0 \\ s_2\epsilon_2 & 1-s_2 \end{bmatrix}.$$

The bulk Finite Difference scheme comes reads

$$\mathsf{m}_1((n+1)\Delta t, x) = ((2-s_2)\mathsf{S}(\mathsf{x}_1) + s_2\epsilon_2\mathsf{A}(\mathsf{x}_1))\mathsf{m}_1(n\Delta t, x) + (s_2-1)\mathsf{m}_1((n-1)\Delta t, x), \tag{10.7}$$

for $n \in [\![Q, +\infty[\![$ and $x \in \Delta x\mathbb{Z}$. This is a Lax-Friedrichs scheme when $s_2 = 1$ —which is first-order consistent with the transport equation at velocity $\lambda\epsilon_2$—and a leap-frog scheme when $s_2 = 2$, which is second-order consistent. Thus, to approximate the solution of (10.1) by $\mathsf{m}_1 \approx u$, the choice of equilibrium is $\epsilon_2 = V/\lambda$. The bulk Finite Difference scheme (10.7) is multi-step with $Q = 1$ when $s_2 \neq 1$: in this case, one needs to specify one initialisation scheme, which is

$$\mathsf{m}_1(\Delta t, x) = (\mathsf{S}(\mathsf{x}_1) + s_2\epsilon_2\mathsf{A}(\mathsf{x}_1))\mathsf{m}_1(0, x) + (1-s_2)\mathsf{A}(\mathsf{x}_1)\mathsf{m}_2(0, x), \qquad x \in \Delta x\mathbb{Z}.$$

We see that both the choice of the conserved moment $\mathsf{m}_1(0,\cdot)$ and the non-conserved moment $\mathsf{m}_2(0,\cdot)$ with respect to u° determine the initial scheme. Unsurprisingly, this scheme coincides with the bulk scheme when $s_2 = 1$.

10.1.5 Overall scheme

The bulk Finite Difference scheme supplemented by the initialisation schemes reads as in Algorithm 7. We stress

Algorithm 7 Corresponding Finite Difference scheme.

- Given $\mathbf{m}(0, \boldsymbol{x})$ for every $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$.
- **Initialisation schemes**. For $n \in [\![1, Q]\!]$

$$\mathsf{m}_1(n\Delta t, \boldsymbol{x}) = (\mathbf{E}^n\mathbf{m})_1(0, \boldsymbol{x}), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d. \tag{10.8}$$

- **Corresponding bulk Finite Difference scheme**. For $n \in [\![Q, +\infty[\![$

$$\mathsf{m}_1((n+1)\Delta t, \boldsymbol{x}) = -\sum_{k=q-Q-1}^{q-1} \varphi_k \mathsf{m}_1((n+k+1-q)\Delta t, \boldsymbol{x}), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d. \tag{10.9}$$

once more that Algorithm 7 is the corresponding scheme of Algorithm 6 in the sense that they issue the same discrete dynamics of the conserved moment m_1 approximating u, see Figure 10.1. Of course, the non-conserved moments $\mathsf{m}_2, \dots, \mathsf{m}_q$ have been eliminated, at the price of handling a multi-step Finite Difference scheme. They still remain in the initialisation (*cf.* Example 10.1.1), giving a first intuition of why we claimed that non-physical modes—associated with non-conserved moments—play a role in this topic.

Remark 10.1.1. *It is worthwhile observing that even if the initialisation schemes (10.8) are considered here close to the initial time,* i.e. *for $n \in [\![1, Q]\!]$, feeding the bulk Finite Difference scheme (10.9), they also represent the action of the lattice Boltzmann scheme through its evolution operator $\mathbf{E}$ away from the initial time, that is, when $n > Q$. In the sequel, we shall employ the following nomenclature:*

- *"**initialisation schemes**", to indicate (10.8) for $n \in [\![1, Q]\!]$;*
- *"**starting schemes**", to indicate (10.8) for any $n \in \mathbb{N}^*$.*

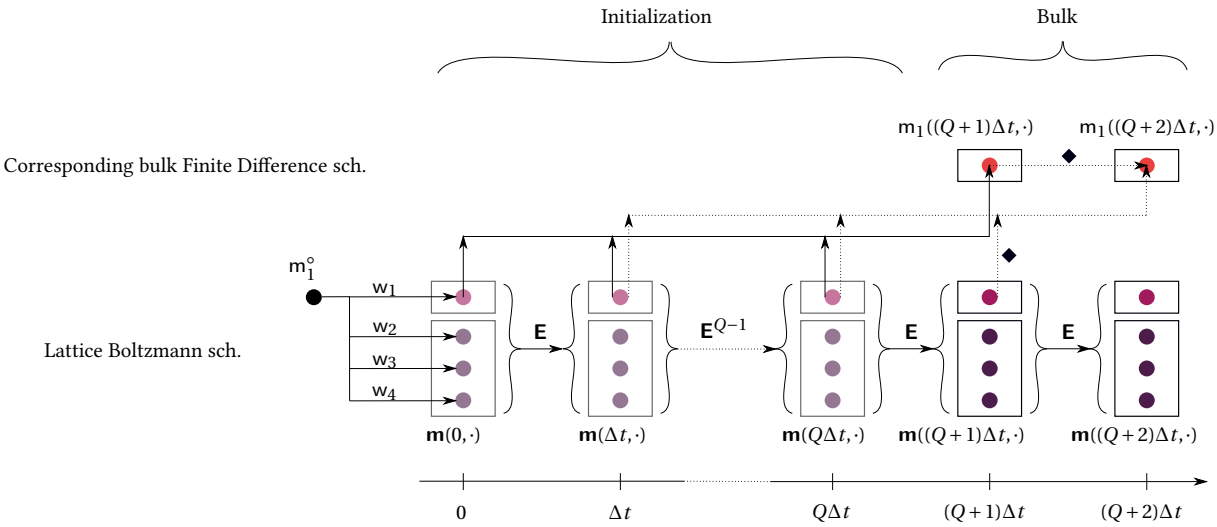


Figure 10.1: Illustration of the way of working of the lattice Boltzmann scheme (bottom) and the bulk Finite Difference scheme (top). The former acts both on the conserved (light violet) and the non-conserved (dark violet) moments. The latter implies only the conserved moment, drawn in light violet in the initialisation layer and in red in the bulk. Remark that to compute the conserved moment for the bulk Finite Difference scheme at time $(Q+2)\Delta t$, one can either rely on the information at time $(Q+1)\Delta t$ in light violet (from the lattice Boltzmann scheme) or on the one in red (from the Finite Difference scheme), as highlighted by the symbol ◆. This holds because these quantities are equal for any time step in the bulk for they stem from a common initialisation process. Partial transparency is used to denote the initialisation steps.

Hence, the initialisation schemes are a proper subset of the starting schemes. Indeed, in Section 10.2 and Section 10.3, we shall also consider the behaviour of (10.8) for $n > Q$, aiming at analysing the agreement between the behaviour of the numerical schemes inside the initial layer and the one purely in the bulk. This idea of matching is reminiscent of the singularly perturbed dynamical systems, see [O'Malley, 1991, Bender et al., 1999].

10.2 Modified equation analysis of the initial conditions under acoustic scaling

In this part of the work, we propose a modified equation analysis for the initialisation schemes, since this procedure gives the equations which carefully describe the real dynamics of the schemes at the desired order of accuracy. The study of the consistency of the initialisation schemes is crucial especially when one wants to reach high-order accuracy. For the overall method, [Strikwerda, 2004, Theorem 10.6.2] states that, under acoustic scaling, if the initialisation of a multi-step scheme is obtained using schemes of accuracy $H-1$ in Δx, where H is the accuracy of the multi-step scheme without accounting for the initialisation, then for smooth initial data, the order of accuracy of the multi-step scheme accounting for the initialisation remains H.

10.2.1 Recap on the modified equation in the bulk

The consistency of the bulk Finite Difference scheme (10.9) has been described in Theorem 8.3.1. We adapt it to the linear setting and since there is only one conserved moment, we use a generic function ϕ instead of m_1.

To perform the consistency analysis of the schemes *via* the modified equation [Warming and Hyett, 1974, Strikwerda, 2004, Gustafsson et al., 1995], one practical way of proceeding is to deploy the scheme on smooth functions over $\mathbb{R}\times\mathbb{R}^d$ instead of on grid functions defined over $\Delta t\mathbb{N}\times\Delta x\mathbb{Z}^d$, and use truncated asymptotic equivalents according to Definition 8.3.1. The scaling assumptions the whole Section 10.2 will rely on are—unless further notice—that $\boldsymbol{M}$, $\boldsymbol{S}$ and $\boldsymbol{\epsilon}$ are independent of Δx as $\Delta x \to 0$. Recall the definition of $\boldsymbol{\mathcal{G}}$ given in (8.7).

Theorem 10.2.1: Modified equation of the bulk scheme

Under acoustic scaling, that is, when the lattice velocity $\lambda > 0$ is fixed as $\Delta x \to 0$, the modified equation for the bulk Finite Difference scheme (10.9) is given by

$$\partial_t \phi(t,\boldsymbol{x}) + \lambda\Big(\mathcal{G}_{11} + \sum_{r=2}^{q} \mathcal{G}_{1r}\epsilon_r\Big)\phi(t,\boldsymbol{x}) \\ - \lambda\Delta x \sum_{i=2}^{q}\Big(\frac{1}{s_i} - \frac{1}{2}\Big)\mathcal{G}_{1i}\Big(\mathcal{G}_{i1} + \sum_{r=2}^{q}\mathcal{G}_{ir}\epsilon_r - \Big(\mathcal{G}_{11} + \sum_{r=2}^{q}\mathcal{G}_{1r}\epsilon_r\Big)\epsilon_i\Big)\phi(t,\boldsymbol{x}) = O(\Delta x^2), \tag{10.10}$$

for $(t,\boldsymbol{x}) \in \mathbb{R}_+ \times \mathbb{R}^d$.

Comparing (10.10) and (10.1), the consistency with the equation of the Cauchy problem shall be enforced selecting the components of the lattice Boltzmann scheme such that $\lambda(\mathcal{G}_{11} + \sum_{r=2}^{r=q}\mathcal{G}_{1r}\epsilon_r) = \boldsymbol{V}\cdot\nabla_{\boldsymbol{x}}$. Since we shall employ the expression "at order $O(\Delta x^h)$" in the following discussion, let us specify what we mean, by taking advantage of the claim from Theorem 10.2.1. The terms ∂_t and $\lambda(\mathcal{G}_{11} + \sum_{r=2}^{r=q}\mathcal{G}_{1r}\epsilon_r)$ appear at order $O(\Delta x)$ when the actual proof of Theorem 10.2.1 is done, *cf.* Section 8.4, thus we call them "$O(\Delta x)$ terms". Then, these terms appear at leading order in (10.10) because all the $O(1)$ terms simplify on both sides of the equation. The remaining term $O(\Delta x)$ in (10.10) originally shows at order $O(\Delta x^2)$ and is made up of numerical diffusion.

10.2.2 Linking the discrete initial datum with the one of the continuous Cauchy problem

We now adapt the same techniques to concentrate on the role of the initial data. From the initial datum of the Cauchy problem u°, we consider its point-wise discretisation with a lattice function m_1° such that $\mathsf{m}_1^\circ(\boldsymbol{x}) = u^\circ(\boldsymbol{x})$ for $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$. Coherently with the fact of considering a linear problem and because the equilibria of the non-conserved moments are linear functions of the conserved one through $\boldsymbol{\epsilon}$, a linear initialisation reads

$$\mathbf{m}(0,\boldsymbol{x}) = \mathbf{w}\mathsf{m}_1^\circ(\boldsymbol{x}), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d, \tag{10.11}$$

where $\mathbf{w}$ can be chosen in two different fashions.

- If $\mathbf{w} \in \mathbb{R}^q$ is considered, we obtain what we call "**local initialisation**". However, in order to gain more freedom on the initialisation and achieve desired numerical properties, another choice is possible.
- If $\mathbf{w} \in \mathsf{D}^q$ is considered, we obtain the "**prepared initialisation**", where we allow for an initial rearrangement of the information issued from the initial datum of the Cauchy problem between neighboring sites of the lattice.

It can be observed that the local initialisation is only a particular case of prepared initialisation using constant polynomials, since $\mathbb{R}$ is a sub-ring of D. By allowing $\mathsf{w}_1 \in \mathsf{D}$, we also permit to perform a preliminary modification of the point-wise discretisation of the initial datum (10.2) of the Cauchy problem, which can also be interpreted as an initial filtering of the datum, before assigning it to m_1. For example, when $d = 1$, considering $\mathsf{w}_1 = \mathsf{S}(\mathsf{x}_1)$ yields $\mathsf{m}_1(0,x) = (u^\circ(x-\Delta x) + u^\circ(x+\Delta x))/2$ for every $x \in \Delta x\mathbb{Z}$. Observe that the following developments can be easily adapted to deal with implicit initialisations [Van Leemput et al., 2009] of the form $\mathsf{w}_i\mathsf{m}_i(0,\boldsymbol{x}) = \mathsf{b}_i\mathsf{m}_1^\circ(\boldsymbol{x})$ with $\mathsf{b}_i \in \mathsf{D}$ for $i \in [\![1,q]\!]$.

10.2.3 Modified equations for the initialisation schemes: local initialisation

Let us now compute the modified equations for the starting schemes when a local initialisation is considered. In the general framework, we shall stop at order $O(\Delta x)$ for two reasons. The first one is that we are not aware of any stable lattice Boltzmann scheme which—under acoustic scaling—would be third-order consistent in the bulk with the target equation (10.1) and therefore would call for second-order accurate initialisation schemes. Second, the expressions for higher order terms are excessively involved to be written down in a convenient form as functions of $n \in [\![1,Q]\!]$ for general schemes and possibly large values of Q. Again, this is due to the role played by the

non-physical eigenvalues of $\mathbf{E}$. Still, one more order in the expansion shall be needed to analyze the smooth initialisation proposed by [Van Leemput et al., 2009, Junk and Yang, 2015], as we shall do in Section 10.3 for some particularly simple yet instructive examples and for a more general class of schemes in Section 10.4.

Proposition 10.2.1: Modified equation of the starting schemes with local initialisation

Under acoustic scaling, that is, when $\lambda > 0$ is fixed as $\Delta x \to 0$, considering a local initialisation, *i.e.* $\mathbf{w} \in \mathbb{R}^q$, the modified equations for the starting schemes are, for any $n \in \mathbb{N}^*$

$$\begin{aligned} &\phi(0,\boldsymbol{x}) + n\frac{\Delta x}{\lambda}\partial_t\phi(0,\boldsymbol{x}) + O(\Delta x^2) \\ &= \mathsf{w}_1\phi(0,\boldsymbol{x}) - n\Delta x\Big(\mathcal{G}_{11}\mathsf{w}_1 + \sum_{b=2}^{q}\mathcal{G}_{1b}\mathsf{w}_b + \frac{1}{n}\sum_{b=2}^{q}\mathcal{G}_{1b}(\epsilon_b\mathsf{w}_1 - \mathsf{w}_b)\sum_{r=0}^{n-1}\pi_{n-r}(s_b)\Big)\phi(0,\boldsymbol{x}) + O(\Delta x^2), \end{aligned} \tag{10.12}$$

for $\boldsymbol{x} \in \mathbb{R}^d$, where $\pi_r(X) = 1-(1-X)^r$ for $r \in \mathbb{N}$.

Proof. We start by describing the particular structure of the powers of collision matrix $\boldsymbol{K}$. It is straightforward to see that we obtain an upper-triangular matrix with

$$\boldsymbol{K}^r = \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ \pi_r(s_2)\epsilon_2 & (1-s_2)^r & 0 & & & \vdots \\ \pi_r(s_3)\epsilon_3 & 0 & (1-s_3)^r & \ddots & & \vdots \\ \pi_r(s_4)\epsilon_4 & 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ \pi_r(s_q)\epsilon_q & 0 & 0 & \cdots & 0 & (1-s_q)^r \end{bmatrix}, \qquad r \in \mathbb{N}^*, \tag{10.13}$$

where the polynomials π_r are defined recursively as $\pi_0(X) := 0$ and $\pi_{r+1}(X) := X + (1-X)\pi_r(X)$ for $r \in \mathbb{N}$. Therefore $\pi_r(X) = 1-(1-X)^r$ for $r \in \mathbb{N}$. The starting schemes read

$$\mathsf{z}^n\mathsf{m}_1(0,\boldsymbol{x}) = (\mathbf{E}^n\mathbf{w})_1\mathsf{m}_1^\circ(\boldsymbol{x}), \qquad n \in \mathbb{N}^*, \quad \boldsymbol{x} \in \Delta x\,\mathbb{Z}^d. \tag{10.14}$$

Concerning the time shifts on the left hand side of (10.14), we have $\mathsf{z}^n \asymp \exp(n\Delta x/\lambda\partial_t) = 1 + n\Delta x/\lambda\partial_t + O(\Delta x^2)$ for $n \in \mathbb{N}$. For the right hand side of (10.14), we have that $\mathbf{E} \asymp \boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{T}}\boldsymbol{K}$ where $\mathbf{T} \asymp \boldsymbol{\mathcal{T}} = \exp(-\Delta x\boldsymbol{\mathcal{G}}) = \boldsymbol{I} - \Delta x\boldsymbol{\mathcal{G}} + O(\Delta x^2)$, see Lemma 8.4.2, and for $n \in \mathbb{N}^*$

$$\begin{aligned} \boldsymbol{\mathcal{E}}^n &= (\boldsymbol{\mathcal{E}}^{(0)} + \Delta x\boldsymbol{\mathcal{E}}^{(1)} + O(\Delta x^2))^n \\ &= (\boldsymbol{\mathcal{E}}^{(0)})^n + \Delta x\sum\{\text{permutations of } \boldsymbol{\mathcal{E}}^{(0)} \ (n-1 \text{ times}) \text{ and } \boldsymbol{\mathcal{E}}^{(1)} \ (\text{once})\} + O(\Delta x^2) \\ &= (\boldsymbol{\mathcal{E}}^{(0)})^n + \Delta x\sum_{r=0}^{r=n-1}(\boldsymbol{\mathcal{E}}^{(0)})^r\boldsymbol{\mathcal{E}}^{(1)}(\boldsymbol{\mathcal{E}}^{(0)})^{n-1-r} + O(\Delta x^2) = \boldsymbol{K}^n - \Delta x\sum_{r=0}^{r=n-1}\boldsymbol{K}^r\boldsymbol{\mathcal{G}}\boldsymbol{K}^{n-r} + O(\Delta x^2), \end{aligned} \tag{10.15}$$

where we use the fact that $\boldsymbol{\mathcal{E}}^{(h)} = \boldsymbol{\mathcal{T}}^{(h)}\boldsymbol{K}$ for $h \in \mathbb{N}$. Plugging into (10.14), employing a smooth function ϕ instead of m_1 and m_1° and using the fact that the initialisation is local, we have for $n \in \mathbb{N}^*$

$$\phi(0,\boldsymbol{x}) + n\frac{\Delta x}{\lambda}\partial_t\phi(0,\boldsymbol{x}) + O(\Delta x^2) = (\boldsymbol{K}^n\mathbf{w})_1\phi(0,\boldsymbol{x}) - \Delta x\Big(\sum_{r=0}^{n-1}\boldsymbol{K}^r\boldsymbol{\mathcal{G}}\boldsymbol{K}^{n-r}\mathbf{w}\Big)_1\phi(0,\boldsymbol{x}) + O(\Delta x^2), \qquad \boldsymbol{x} \in \mathbb{R}^d.$$

We have that $(\boldsymbol{K}^n\mathbf{w})_1\phi(0,\boldsymbol{x}) = \mathsf{w}_1\phi(0,\boldsymbol{x})$ thanks to (10.13) and for $j \in [\![1,q]\!]$

$$\begin{aligned} (\boldsymbol{K}^r\boldsymbol{\mathcal{G}}\boldsymbol{K}^{n-r})_{1j} = \sum_{p=1}^{q}\sum_{b=1}^{q}(\boldsymbol{K}^r)_{1p}\mathcal{G}_{pb}(\boldsymbol{K}^{n-r})_{bj} &= \sum_{b=1}^{q}\mathcal{G}_{1b}(\boldsymbol{K}^{n-r})_{bj} \\ &= \mathcal{G}_{11}\delta_{1j} + \sum_{b=2}^{q}\mathcal{G}_{1b}(\pi_{n-r}(s_b)\epsilon_b\delta_{1j} + (1-s_b)^{n-r}\delta_{bj}). \end{aligned}$$

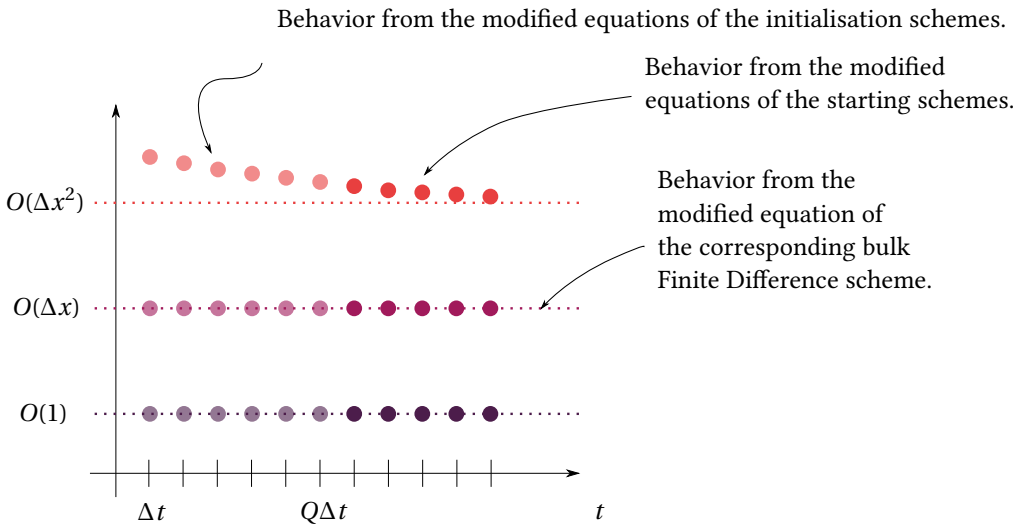


Figure 10.2: Example of behaviour of the inner expansion (dots, concerning the starting schemes) and the outer expansion (dashed lines, relative to the bulk Finite Difference scheme) at different orders in Δx for $\Delta x \to 0$.

Therefore for $n \in \mathbb{N}^*$

$$\begin{aligned}\sum_{r=0}^{n-1} (\boldsymbol{K}^r \boldsymbol{\mathcal{G}} \boldsymbol{K}^{n-r} \mathbf{w})_1 &= n\mathcal{G}_{11}\mathsf{w}_1 + \sum_{b=2}^{q} \mathcal{G}_{1b} \sum_{r=0}^{n-1} (\pi_{n-r}(s_b)\epsilon_b + (1-s_b)^{n-r}\mathsf{w}_b) \\ &= n\Big(\mathcal{G}_{11}\mathsf{w}_1 + \sum_{b=2}^{q} \mathcal{G}_{1b}\mathsf{w}_b + \frac{1}{n}\sum_{b=2}^{q} \mathcal{G}_{1b}(\epsilon_b \mathsf{w}_1 - \mathsf{w}_b) \sum_{r=0}^{n-1} \pi_{n-r}(s_b)\Big),\end{aligned}$$

where we have used that by the definition of π_r, $(n - \sum_{r=0}^{r=n-1} \pi_{n-r}(s_b))/\sum_{r=0}^{r=n-1}(1-s_b)^{n-r} = 1$ for every $n \in \mathbb{N}^*$, yielding the claim. □

With Proposition 10.2.1, we can now compare the modified equation for the bulk Finite Difference scheme and the modified equations of the starting schemes, so respectively the dashed lines and the dots in Figure 10.2 at order $O(1)$ and $O(\Delta x)$.

10.2.4 Consistency of the initialisation schemes: local initialisation

The agreement between the terms at these two orders takes place under the following conditions.

Corollary 10.2.1: Consistency of the starting schemes with local initialisation

Under acoustic scaling, that is, when $\lambda > 0$ is fixed as $\Delta x \to 0$, considering a local initialisation, *i.e.* $\mathbf{w} \in \mathbb{R}^q$, under the conditions

$$\mathsf{w}_1 = 1, \tag{10.16}$$

$$\text{for} \quad b \in [\![2, q]\!], \quad \text{if} \quad \mathcal{G}_{1b} \neq 0, \quad \text{then} \quad \mathsf{w}_b = \epsilon_b, \tag{10.17}$$

where $\boldsymbol{\epsilon}$ are the equilibrium coefficients, the starting schemes are consistent with the modified equation (10.10) of the bulk Finite Difference scheme at order $O(\Delta x)$. Moreover, the initial datum feeding the bulk Finite Difference scheme and the starting schemes is consistent with the initial datum (10.2) of the Cauchy problem.

The first condition (10.16) implies that the initial datum for m_1, used both by the starting schemes and the bulk Finite Difference scheme, is left untouched compared to the one of the Cauchy problem. The second condition (10.17) is expected: for the non-conserved moments involved in the modified equation (10.10) at leading order, we need to consider the initial datum at equilibrium. It is also to observe that this requirement does not *a priori* fix

all the initialisation parameters, contrarily to Example 10.1.1, because some of them can affect only higher orders in the developments, *i.e.* $\mathcal{G}_{1b} = 0$ for some $b \in [\![1, q]\!]$.

Proof of Corollary 10.2.1. The proof proceeds order-by-order in Δx.

- $O(1)$. This order indicates that the initial datum for the conserved moment has to be consistent with the one of the Cauchy problem (10.2). From Proposition 10.2.1, it reads

$$\phi(0, \boldsymbol{x}) = \mathsf{w}_1 \phi(0, \boldsymbol{x}) + O(\Delta x), \qquad n \in \mathbb{N}^*, \quad \boldsymbol{x} \in \mathbb{R}^d, \tag{10.18}$$

  hence we enforce $\mathsf{w}_1 = 1$. Remark that (10.18) is satisfied both for $n \in [\![1, Q]\!]$ and for $n > Q$, that is, both for initialisation schemes and starting schemes. This condition being fulfilled, the next order to check is

$$\partial_t \phi(0, \boldsymbol{x}) + \lambda \Big(\mathcal{G}_{11} + \sum_{b=2}^{q} \mathcal{G}_{1b} \mathsf{w}_b + \frac{1}{n} \sum_{b=2}^{q} \mathcal{G}_{1b} (\epsilon_b - \mathsf{w}_b) \sum_{r=0}^{n-1} \pi_{n-r}(s_b) \Big) = O(\Delta x), \qquad n \in \mathbb{N}^*, \quad \boldsymbol{x} \in \mathbb{R}^d. \tag{10.19}$$

- $O(\Delta x)$. Evaluating the bulk modified equation (10.10) at time $t = 0$ gives

$$\partial_t \phi(0, \boldsymbol{x}) + \lambda \Big(\mathcal{G}_{11} + \sum_{b=2}^{q} \mathcal{G}_{1b} \epsilon_b \Big) \phi(0, \boldsymbol{x}) = O(\Delta x), \qquad \boldsymbol{x} \in \mathbb{R}^d, \tag{10.20}$$

  and trying to match each term with (10.19) yields the condition

$$\text{for} \quad b \in [\![2, q]\!], \quad \text{if} \quad \mathcal{G}_{1b} \neq 0, \quad \text{then} \quad \mathsf{w}_b = \epsilon_b.$$

□

10.2.5 Modified equations for the initialisation schemes: prepared initialisation

Now that the principles concerning the computation of modified equations for the starting schemes and the way of matching terms with the modified equation of the bulk Finite Difference scheme are clarified, we can tackle the case of prepared initialisations.

Proposition 10.2.2: Modified equation of the starting schemes with prepared initialisation

Under acoustic scaling, that is, when $\lambda > 0$ is fixed as $\Delta x \to 0$, considering a prepared initialisation, *i.e.* $\mathbf{w} \in \mathsf{D}^q$, which can be put under the form

$$\mathsf{w}_i = \sum_{\mathfrak{e}} w_{i,\mathfrak{e}} \mathbf{x}^{\mathfrak{e}}, \qquad i \in [\![1, q]\!], \tag{10.21}$$

where the sequences of coefficients $(w_{i,\mathfrak{e}})_{\mathfrak{e}} \subset \mathbb{R}$ are compactly supported, the modified equations for the starting schemes are, for any $n \in \mathbb{N}^*$ and $\boldsymbol{x} \in \mathbb{R}^d$

$$\begin{aligned} \phi(0, \boldsymbol{x}) + n \frac{\Delta x}{\lambda} \partial_t \phi(0, \boldsymbol{x}) + O(\Delta x^2) &= \omega_1^{(0)} \phi(0, \boldsymbol{x}) \\ &- n \Delta x \Big(\mathcal{G}_{11} \omega_1^{(0)} + \sum_{b=2}^{q} \mathcal{G}_{1b} \omega_b^{(0)} + \frac{1}{n} \sum_{b=2}^{q} \mathcal{G}_{1b} (\epsilon_b \omega_1^{(0)} - \omega_b^{(0)}) \sum_{r=0}^{n-1} \pi_{n-r}(s_b) - \frac{1}{n} \omega_1^{(1)} \Big) \phi(0, \boldsymbol{x}) + O(\Delta x^2), \end{aligned}$$

where

$$\omega_i^{(0)} = \sum_{\mathfrak{e}} w_{i,\mathfrak{e}}, \qquad \omega_i^{(1)} = - \sum_{|\mathfrak{n}|=1} \Big(\sum_{\mathfrak{e}} w_{i,\mathfrak{e}} \mathfrak{e}^{\mathfrak{n}} \Big) \partial_{\boldsymbol{x}}^{\mathfrak{n}}, \qquad i \in [\![1, q]\!],$$

and such that $\mathsf{w}_i \asymp \omega_i^{(0)} + \Delta x \omega_i^{(1)} + O(\Delta x)$ and $\pi_r(X) = 1 - (1 - X)^r$ for $r \in \mathbb{N}$.

Proof. The asymptotic equivalent of the initialisation $\mathbf{w}$ reads

$$\mathsf{w}_i \asymp \omega_i = \sum_{\mathfrak{e}} w_{i,\mathfrak{e}} - \Delta x \sum_{|\mathfrak{n}|=1} \Big(\sum_{\mathfrak{e}} w_{i,\mathfrak{e}} \mathfrak{e}^{\mathfrak{n}} \Big) \partial_{\boldsymbol{x}}^{\mathfrak{n}} + O(\Delta x^2), \qquad i \in [\![1, q]\!].$$

Using the Cauchy product between formal series, we have $\mathbf{E}^n \mathbf{w} \asymp \boldsymbol{\mathcal{E}}^n \boldsymbol{\omega} = (\boldsymbol{\mathcal{E}}^n)^{(0)} \boldsymbol{\omega}^{(0)} + \Delta x ((\boldsymbol{\mathcal{E}}^n)^{(1)} \boldsymbol{\omega}^{(0)} + (\boldsymbol{\mathcal{E}}^n)^{(0)} \boldsymbol{\omega}^{(1)}) + O(\Delta x^2)$ for $n \in \mathbb{N}^*$. The $O(\Delta x)$ term in the previous expansion is made up of two contributions. The first one is $(\boldsymbol{\mathcal{E}}^n)^{(1)} \boldsymbol{\omega}^{(0)}$ and is not influenced by the "prepared" character of the initialisation, because it was also present for the local initialisation. The second one is inherent to the prepared initialisation. The result comes from the very same computations as Proposition 10.2.1. □

10.2.6 Consistency of the initialisation schemes: prepared initialisation

Corollary 10.2.2: Consistency of the starting schemes with prepared initialisation

Under acoustic scaling, that is, when $\lambda > 0$ is fixed as $\Delta x \to 0$, considering a prepared initialisation, *i.e.* $\mathbf{w} \in \mathsf{D}^q$, with (10.21), under the conditions

$$\sum_{\mathfrak{e}} w_{1,\mathfrak{e}} = 1, \tag{10.22}$$

$$\text{for every} \quad |\mathfrak{n}| = 1, \quad \sum_{\mathfrak{e}} w_{1,\mathfrak{e}} \mathfrak{e}^{\mathfrak{n}} = 0, \tag{10.23}$$

$$\text{for} \quad b \in [\![2, q]\!], \quad \text{if} \quad \mathcal{G}_{1b} \neq 0, \quad \text{then} \quad \sum_{\mathfrak{e}} w_{b,\mathfrak{e}} = \epsilon_b, \tag{10.24}$$

the starting schemes are consistent with the modified equation (10.10) of the bulk Finite Difference scheme at order $O(\Delta x)$. Moreover, the initial datum feeding the bulk Finite Difference scheme and the starting schemes is consistent with the initial datum (10.2) of the Cauchy problem up to order $O(\Delta x^2)$.

Condition (10.22) is the analogue of (10.16). However, since the initialisation of the conserved moment can also be prepared, an additional condition (10.23) has to be taken into account. This guarantees, in particular, that the initial datum of the Cauchy problem used for m_1 is not perturbed by some drift term at order $O(\Delta x)$. This is useful because of the multi-step nature of the bulk Finite Difference scheme (10.9), which shall also be fed with (10.11). Finally, (10.24) has to be compared with (10.17). This condition maintains that the non-conserved moments participating to the consistency at leading order have to be chosen—at leading order—at equilibrium.

Proof of Corollary 10.2.2. Proceeding order-by-order in Δx, we obtain:

- $O(1)$. The dominant order in the analogous of (10.12). Hence the consistency with the datum of the Cauchy problem reads $\omega_1^{(0)} = \sum_{\mathfrak{e}} w_{1,\mathfrak{e}} = 1$.
- $O(\Delta x)$. We see that now, there is the additional term associated with $\omega_1^{(1)}$ corresponding to a drift term in the initialisation of the conserved moment. In general, we now have wider possibilities in terms of how initialise, still remaining consistent with the modified equation of the bulk Finite Difference scheme at the desired order, at least for the initialisation schemes (*i.e.* $n \in [\![1, Q]\!]$). Indeed, it is sufficient to enforce that

$$\mathcal{G}_{11} + \sum_{b=2}^{q} \mathcal{G}_{1b} \omega_b^{(0)} + \frac{1}{n} \sum_{b=2}^{q} \mathcal{G}_{1b} (\epsilon_b - \omega_b^{(0)}) \sum_{r=0}^{n-1} \pi_{n-r}(s_b) - \frac{1}{n} \omega_1^{(1)} = \mathcal{G}_{11} + \sum_{b=2}^{q} \mathcal{G}_{1b} \epsilon_b.$$

  Occasionally, for some $n \in [\![1, Q]\!]$, the previous inequality can be satisfied even if $\omega_1^{(1)} \neq 0$, see examples in Section 10.3. However, we are interested in enforcing it for every for every $n \in \mathbb{N}^*$—that is—for all starting schemes. This comes, as previously claimed, from the multi-step nature of the bulk Finite Difference scheme: we have to ensure that the order of consistency with the initial datum (10.2) of the Cauchy problem is high enough not to lower the overall order of the method. Hence, suppressing the drift term for the conserved

moment, thus enforcing $\omega_1^{(1)} = 0$, we have

$$\text{for every} \quad |\mathfrak{n}| = 1, \quad \sum_{\mathfrak{e}} w_{1,\mathfrak{e}} \mathfrak{e}^{\mathfrak{n}} = 0, \quad \text{and} \quad \text{for} \quad b \in [\![2, q]\!], \quad \text{if} \quad \mathcal{G}_{1b} \neq 0, \quad \text{then} \quad \sum_{\mathfrak{e}} w_{b,\mathfrak{e}} = \epsilon_b.$$

□

10.2.7 Initialisation schemes *versus* starting schemes

Before proceeding to some numerical illustrations, we point out important facts concerning the match of terms between the bulk Finite Difference scheme and the initialisation schemes/starting schemes.

Proposition 10.2.3: Control on the initialisation schemes leads control on the starting schemes

Let $H \in \mathbb{N}^*$. Assume that

- $\omega_1^{(0)} = 1$ and $\omega_1^{(h)} = 0$ for $h \in [\![1, H]\!]$.
- The modified equations of the initialisation schemes ((10.8) for $n \in [\![1, Q]\!]$) match the one of the bulk Finite Difference scheme (10.9) at any order $h \in [\![1, H]\!]$.

Then, the modified equations of the starting schemes ((10.8) for $n > Q$) match the one of the bulk Finite Difference scheme (10.9) at any order $h \in [\![1, H]\!]$.

Proposition 10.2.3 does not provide indications on how to equate the order at $h \in [\![1, H]\!]$—*i.e.* how to fulfill its assumptions— contrarily to what Corollary 10.2.1 and Corollary 10.2.2 do for $H = 1$. Again, this is due to the fact that the general expression of the asymptotic expansion of $(\mathbf{E}^n \mathbf{w})_1$ can quickly become messy as the considered order increases. Still, Proposition 10.2.3 claims that if one is able to match the modified equation of the initialisation schemes with the one of the bulk Finite Difference scheme until a given order H (as we shall do for specific schemes in Section 10.3 and Section 10.4 with $H = 2$), then this guarantees the same property on the starting schemes. Otherwise said—referring to Figure 10.2—if one is able to ensure that the terms represented by the dots lie on the corresponding dashed line for $n \in [\![1, Q]\!]$, then one will be sure that these dots will lie on the very same line for any $n \in \mathbb{N}^*$. This result seems intuitively reasonable by virtue of the Cayley-Hamilton theorem, which allows to recast any power $\mathbf{E}^n$ for $n \geq Q$ as combination of $\boldsymbol{I}, \mathbf{E}, \ldots, \mathbf{E}^{Q-1}$.

Proof of Proposition 10.2.3. Let us consider $d = 1$ for the sake of notation: for $d > 1$, the multi-index notation would suffice. Consider a one-step linear Finite Difference scheme on the lattice function u, under the form $\mathsf{z}\mathsf{u}(t,x) = \mathsf{g}_1 \mathsf{u}(t,x)$ for $(t,x) \in \Delta t\mathbb{N} \times \Delta x\mathbb{Z}$, where $\mathsf{g}_1 \in \mathsf{D}$. This can rewritten using the Fourier transform in space, that is

$$\mathsf{z}\hat{\mathsf{u}}(t,\theta) = \hat{\mathsf{g}}_1(\theta)\hat{\mathsf{u}}(t,\theta), \qquad (t,\theta) \in \Delta t\mathbb{N} \times [-\pi, \pi]. \tag{10.25}$$

The frequency-dependent eigenvalue $\hat{\mathsf{g}}_1(\theta) \in \mathbb{C}$ shall be a Laurent polynomial in the indeterminate $e^{i\theta}$ and encodes both the stability features of the method, for every $\theta \in [-\pi, \pi]$ and the consistency features, in the low-frequency limit $|\theta| \ll 1$. In particular, to be consistent with an equation of the form (10.1) with a first-order derivative in time, one can easily see that

$$\hat{\mathsf{g}}_1(\theta) = 1 + O(|\theta|) \qquad \text{in the limit} \quad |\theta| \ll 1. \tag{10.26}$$

Applying the scheme (10.25) $n \in \mathbb{N}^*$ times provides a sort of multi-step scheme which we shall compare to the starting schemes (10.8)

$$\mathsf{z}^n \hat{\mathsf{u}}(t,\theta) = \hat{\mathsf{g}}_1(\theta)^n \hat{\mathsf{u}}(t,\theta), \qquad (t,\theta) \in \Delta t\mathbb{N} \times [-\pi, \pi], \tag{10.27}$$

with associated amplification polynomial

$$\hat{\tilde{\Phi}}(\theta, \mathsf{z}) = \mathsf{z}^n - \hat{\mathsf{g}}_1(\theta)^n = (\mathsf{z} - \hat{\mathsf{g}}_1(\theta)) \sum_{r=0}^{n-1} \hat{\mathsf{g}}_1(\theta)^r \mathsf{z}^{n-1-r} = (\mathsf{z} - \hat{\mathsf{g}}_1(\theta)) \prod_{r=2}^{n} (\mathsf{z} - \hat{\tilde{\mathsf{g}}}_r(\theta)), \tag{10.28}$$

having roots $\hat{\mathsf{g}}_1 = \hat{\mathsf{g}}_1$ and $\hat{\mathsf{g}}_2, \dots, \hat{\mathsf{g}}_n$ (recall that $\mathbb{C}$ is an algebraically closed field). By differentiating the amplification polynomial (10.28) using the rule for the derivative of a product, we get

$$\frac{\mathrm{d}\hat{\Phi}(\theta,\mathsf{z})}{\mathrm{d}\mathsf{z}} = n\mathsf{z}^{n-1} = \prod_{r=2}^{n}(\mathsf{z}-\hat{\mathsf{g}}_r(\theta)) + (\mathsf{z}-\hat{\mathsf{g}}_1(\theta))\sum_{r=2}^{n}\prod_{\substack{p=2\\p\neq r}}^{n}(\mathsf{z}-\hat{\mathsf{g}}_p(\theta)).$$

Taking $\mathsf{z}=1$ in the limit $|\theta| \ll 1$ gives $0 \neq n = \prod_{r=2}^{r=n}(1-\hat{g}_r^{(0)})$ thanks to (10.26), where $\hat{\mathsf{g}}_r(\theta) = \hat{g}_r^{(0)} + O(|\theta|)$, thus all the other eigenvalues $\hat{\mathsf{g}}_2, \dots, \hat{\mathsf{g}}_n$ are not equal to one for small frequencies and thus are not linked with consistency, but are merely numerical eigenvalues. The only which matters is $\hat{\mathsf{g}}_1 = \hat{\mathsf{g}}_1$, thus the scheme (10.27) with amplification polynomial (10.28) has the same modified equations as (10.25). An alternative way of seeing this is to use the approach from the proof of [Carpentier et al., 1997, Proposition 1] , which aims at automatically handling the "reinjection" of previous orders in the expansions to eliminate time derivatives above first order. Inserting the asymptotic equivalent $\exp(n\Delta t\partial_t) \asymp \mathsf{z}^n$ into (10.27) using a smooth "test" function $\hat{\phi}$ gives $\exp(n\Delta t\partial_t)\hat{\phi}(t,\xi) = \hat{\mathsf{g}}_1(\xi\Delta x)^n\hat{\phi}(t,\xi)$ for $(t,\xi) \in \mathbb{R}_+ \times \mathbb{R}$, which means that if we do not want $\hat{\phi}$ to trivially vanish, we must enforce the formal identity $\exp(n\Delta t\partial_t) = \hat{\mathsf{g}}_1(\xi\Delta x)^n$. Since the exponential is bijective close to zero (here we are considering the limit $|\xi\Delta x| \ll 1$), we can take the logarithm to yield:

$$\partial_t = \frac{n}{n}\underbrace{\frac{1}{\Delta t}}_{=\lambda/\Delta x}\log(\hat{\mathsf{g}}_1(\xi\Delta x)),$$

which is thus independent of n.

Differently, a $(Q+2)$-stages Finite Difference scheme, like the bulk Finite Difference scheme (10.9), has associated amplification polynomial

$$\hat{\Phi}(\xi\Delta x,\mathsf{z}) := \frac{1}{\mathsf{z}^{q-Q-1}}\det(\mathsf{z}\boldsymbol{I}-\hat{\mathbf{E}}(\xi\Delta x)) = \mathsf{z}^{Q+1} + \sum_{k=0}^{Q}\hat{\varphi}_{k+q-Q-1}(\xi\Delta x)\mathsf{z}^k = \prod_{r=1}^{Q+1}(\mathsf{z}-\hat{\mathsf{g}}_r(\xi\Delta x)). \tag{10.29}$$

Out of the roots in (10.29), we shall number the (unique) eigenvalue providing the modified equation (10.10), *i.e.* such that (10.26) holds, by $\hat{\mathsf{g}}_1$, see Theorem 7.7.1. This is the amplification factor of the so-called "pseudo-scheme" [Strikwerda, 2004, Chapter 10]. It is not associated with a Finite Difference operator, thus it is not a Laurent polynomial in the indeterminate $e^{-i\theta}$ (recall Remark 7.5.1 on the eigenvalues of matrices with entries in D or $\hat{\mathsf{D}}$). Still, it behaves essentially as a Finite Difference scheme. Furthermore, the higher-order terms in the modified equation of the bulk Finite Difference scheme stem from $\hat{\mathsf{g}}_1(\xi\Delta x) = 1 + \sum_{h=1}^{h=H}(\xi\Delta x)^h\hat{g}_1^{(h)} + O(|\xi\Delta x|^{H+1})$ in the limit $|\xi\Delta x| \ll 1$. The initialisation schemes read

$$\mathsf{z}^n\hat{\mathsf{m}}_1(0,\xi\Delta x) = \underbrace{(\hat{\mathbf{E}}(\xi\Delta x)^n\hat{\mathbf{w}}(\xi\Delta x))_1}_{=:\hat{\mathsf{g}}^{[n]}(\xi\Delta x)}\hat{\mathsf{m}}_1^{\circ}(\xi\Delta x), \qquad n \in [\![1,Q]\!], \tag{10.30}$$

with $\xi \in [-\pi/\Delta x, \pi/\Delta x]$. Using the assumption that $\omega_1^{(0)} = 1$, the proof of Proposition 10.2.2 naturally entails that $\hat{\mathsf{g}}^{[n]}(\xi\Delta x) = 1 + O(|\xi\Delta x|)$ for $|\xi\Delta x| \ll 1$. Comparing (10.27) and (10.30), we cannot employ the same trick without a deeper discussion. We have respectively

$$\partial_t = \frac{\lambda}{\Delta x}\log(\hat{\mathsf{g}}_1(\xi\Delta x)) \qquad \text{and} \qquad \partial_t = \frac{1}{n}\frac{\lambda}{\Delta x}\log(\hat{\mathsf{g}}^{[n]}(\xi\Delta x)), \qquad n \in [\![1,Q]\!],$$

where the first equation comes from the modified equation of the bulk Finite Difference scheme and the second one from (10.30). Since the initialisation schemes and the bulk Finite Difference scheme have the same modified equations up to order H, then we have, in the limit $|\xi\Delta x| \ll 1$

$$n\log(\hat{\mathsf{g}}_1(\xi\Delta x)) = \log(\hat{\mathsf{g}}_1(\xi\Delta x)^n) = \log(\hat{\mathsf{g}}^{[n]}(\xi\Delta x)) + O(|\xi\Delta x|^{H+1}), \qquad n \in [\![1,Q]\!],$$

hence we deduce that $\hat{\mathsf{g}}^{[n]}(\xi\Delta x) = \hat{\mathsf{g}}_1(\xi\Delta x) + O(|\xi\Delta x|^{H+1})$ for $n \in [\![1,Q]\!]$. To finish the proof, we now consider (10.8)

for $n = Q+1$

$$\mathsf{z}^{Q+1}\hat{\mathsf{m}}_1(0,\xi\Delta x) = -\sum_{k=0}^{Q}\hat{\varphi}_{k+q-Q-1}(\xi\Delta x)\mathsf{z}^k\hat{\mathsf{m}}_1(0,\xi\Delta x) \tag{10.31}$$

$$= \underbrace{(\hat{\mathbf{E}}(\xi\Delta x)^{Q+1}\hat{\mathbf{w}}(\xi\Delta x))_1}_{=:\hat{\mathsf{g}}^{[Q+1]}(\xi\Delta x)}\hat{\mathsf{m}}_1^{\circ}(\xi\Delta x). \tag{10.32}$$

We compute the modified equation of (10.32), yielding the thesis, by using (10.31). We have

$$\begin{aligned}\mathsf{z}^{Q+1}\hat{\mathsf{m}}_1(0,\xi\Delta x) &= -\sum_{k=1}^{Q}\hat{\varphi}_{k+q-Q-1}(\xi\Delta x)\mathsf{z}^k\hat{\mathsf{m}}_1(0,\xi\Delta x) - \hat{\varphi}_{q-Q-1}(\xi\Delta x)\hat{\mathsf{w}}_1(\xi\Delta x)\hat{\mathsf{m}}_1^{\circ}(\xi\Delta x)\\ &= -\sum_{k=1}^{Q}\hat{\varphi}_{k+q-Q-1}(\xi\Delta x)\hat{\mathsf{g}}^{[k]}(\xi\Delta x)\hat{\mathsf{m}}_1^{\circ}(\xi\Delta x) - \hat{\varphi}_{q-Q-1}(\xi\Delta x)\hat{\mathsf{w}}_1(\xi\Delta x)\hat{\mathsf{m}}_1^{\circ}(\xi\Delta x).\end{aligned}$$

In the limit $|\xi\Delta x| \ll 1$, we have $\hat{\mathsf{w}}_1(\xi\Delta x) = 1 + O(|\xi\Delta x|^{H+1})$ and $\hat{\mathsf{g}}^{[n]}(\xi\Delta x) = \hat{\mathsf{g}}_1(\xi\Delta x)^n + O(|\xi\Delta x|^{H+1})$ for $n \in [\![1, Q]\!]$, thanks to the assumption on w_1 and to the previous computations. In the limit $|\xi\Delta x| \ll 1$, we have to consider the amplification polynomial

$$\hat{\tilde{\Phi}}(\xi\Delta x, \mathsf{z}) = \mathsf{z}^{Q+1} + \sum_{k=0}^{Q}\hat{\varphi}_{k+q-Q-1}(\xi\Delta x)\hat{\mathsf{g}}_1(\xi\Delta x)^k + O(|\xi\Delta x|^{H+1}) = \mathsf{z}^{Q+1} - \hat{\mathsf{g}}_1(\xi\Delta x)^{Q+1} + O(|\xi\Delta x|^{H+1}),$$

using the fact that $\hat{\mathsf{g}}_1$ is a root of (10.29). We are therefore, up to terms $O(|\xi\Delta x|^{H+1})$, in the same setting as (10.27) and (10.28), hence with the usual trick, we gain

$$\partial_t = \frac{Q+1}{Q+1}\frac{\lambda}{\Delta x}\log(\hat{\mathsf{g}}_1(\xi\Delta x)) + O(|\xi\Delta x|^{H+1}),$$

hence also that $\hat{\mathsf{g}}^{[Q+1]}(\xi\Delta x) = \hat{\mathsf{g}}_1(\xi\Delta x)^{Q+1} + O(|\xi\Delta x|^{H+1})$. This concludes the proof. The case $n > Q+1$ is done analogously. □

A second result states that there is little interest in considering the formal limit $n \to +\infty$ in the modified equations of the starting schemes.

Proposition 10.2.4: Long-time behavior: limits for $n \to +\infty$

Assume that the scheme is stable according to Theorem 7.7.3, meaning that the roots of the amplification polynomial $\mathsf{z}^{Q+1-q}\det(\mathsf{z}\boldsymbol{I} - \hat{\mathbf{E}})$ fulfill Theorem 7.7.3. Then:

- If $|1 - s_i| < 1$ for $i \in [\![2, q]\!]$, or equivalently $s_i \in]0, 2[$ for $i \in [\![2, q]\!]$, then the modified equations of the starting schemes in the formal long-time limit $n \to +\infty$ coincide at any order with the one of the bulk Finite Difference scheme.
- If it exists $\tilde{i} \in [\![2, q]\!]$ such that $|1 - s_{\tilde{i}}| = 1$, thus equivalently $s_{\tilde{i}} = 2$. Let $H \in \mathbb{N}^*$. Provided that $\omega_1^{(h)} = 0$ for $h \in [\![1, H]\!]$ and the Q modified equations of the initialisation schemes coincide with the one of the bulk Finite Difference scheme at any order $h \in [\![1, H]\!]$, then the modified equations of the starting schemes in the formal long-time limit $n \to +\infty$ coincide at any order $h \in [\![1, H+1]\!]$ with the one of the bulk Finite Difference scheme.

This means that the effect of the initialisation decays, provided that the initial filtering on the datum (10.2) preserves the initial datum at leading order and that either the parasitic modes damp out in time, or if the parasitic modes are oscillatory, the initialisation schemes are accurate enough. The situation is the one depicted in Figure 10.2, where the dots asymptotically reach the dashed lines. Let us point out that the assumption concerning stability may not be optimal, in the sense that we can find unstable schemes (for example, violating the CFL condition, but not having relaxation parameters exceeding 2) for which the modified equations of the starting schemes asymptotically reach those of the bulk Finite Difference scheme. However, this schemes are practically useless.

Proof of Proposition 10.2.4. Before proceeding, let us insist on the fact that the Cayley-Hamilton theorem entails that the amplification factors of the starting schemes which are not initialisation schemes can be computed in two ways, which read $\hat{\mathsf{g}}^{[n]} = \boldsymbol{e}_1^{\mathsf{t}}\hat{\mathbf{E}}^n\hat{\mathbf{w}} = \boldsymbol{e}_1^{\mathsf{t}}\hat{\mathbf{Q}}^{n-Q}[\hat{\mathsf{g}}^{[Q]},\dots,\hat{\mathsf{g}}^{[1]},\hat{\mathsf{w}}_1]^{\mathsf{t}}$, for $n \geq Q+1$. In this expression, $\hat{\mathbf{Q}}$ is the companion matrix associated with the amplification polynomial $\mathsf{z}^{Q+1-q}\det(\mathsf{z}\boldsymbol{I} - \hat{\mathbf{E}})$.

Let us formulate a preliminary remark: we consider formal series in the limit $|\xi\Delta x| \ll 1$. Therefore, the possibility of diagonalise $\hat{\mathbf{Q}}(\xi\Delta x)$ in this limit—or alternatively being obliged to deal with a true Jordan canonical form—is determined by $\hat{\mathbf{Q}}(0)$, *i.e.* the leading-order term in the formal series. We assume—without loss of generality—that all $\hat{\mathsf{g}}_r(0)$ for $r \in [\![1, Q+1]\!]$ are simple, even those strictly inside the unit circle, so that we can diagonalise the companion matrix in the desired limit. If this does not hold, for example for a SRT (or BGK) scheme, one can easily go through the same proof using the well-known expression for the powers of Jordan blocks.

Let us start the proof. Let $\hat{\mathbf{V}} = \hat{\mathbf{V}}(\xi\Delta x)$ be the Vandermonde matrix associated with $\hat{\mathsf{g}}_1 = \hat{\mathsf{g}}_1(\xi\Delta x),\dots,\hat{\mathsf{g}}_{Q+1} = \hat{\mathsf{g}}_{Q+1}(\xi\Delta x)$, the roots of $\mathsf{z}^{Q+1-q}\det(\mathsf{z}\boldsymbol{I} - \hat{\mathbf{E}})$. It is well-known that this Vandermonde matrix diagonalises the companion matrix $\hat{\mathbf{Q}}(\xi\Delta x)$, thus for $n \geq Q+1$

$$\begin{aligned}\hat{\mathsf{g}}^{[n]}(\xi\Delta x) &= \boldsymbol{e}_1^{\mathsf{t}}\hat{\mathbf{Q}}(\xi\Delta x)^{n-Q}[\hat{\mathsf{g}}^{[Q]}(\xi\Delta x),\dots,\hat{\mathsf{g}}^{[1]}(\xi\Delta x),\hat{\mathsf{w}}_1(\xi\Delta x)]^{\mathsf{t}} \\ &= \boldsymbol{e}_1^{\mathsf{t}}\hat{\mathbf{V}}(\xi\Delta x)\mathrm{diag}(\hat{\mathsf{g}}_1(\xi\Delta x)^{n-Q},\dots,\hat{\mathsf{g}}_{Q+1}(\xi\Delta x)^{n-Q})\hat{\mathbf{V}}(\xi\Delta x)^{-1}[\hat{\mathsf{g}}^{[Q]}(\xi\Delta x),\dots,\hat{\mathsf{g}}^{[1]}(\xi\Delta x),\hat{\mathsf{w}}_1(\xi\Delta x)]^{\mathsf{t}}.\end{aligned} \tag{10.33}$$

The idea of the proof is that the amplification factors associated with the initialisation schemes form an approximation of the eigenvector of $\hat{\mathbf{Q}}(\xi\Delta x)$ relative to the consistency eigenvalue $\hat{\mathsf{g}}_1$, so that the power iteration (10.33) converges for $n \to +\infty$ up to some order. Up to a re-ordering of the non-conserved moments—in order to start with those which do not relax on the equilibrium, for notational ease—the lower-triangular structure of the collision matrix $\boldsymbol{K}$ entails that $\hat{\mathsf{g}}_r(0) = 1 - s_r$ for $r \in [\![2, Q+1]\!]$. Moreover, we have that $\hat{\mathsf{g}}_1(0) = 1$.

- Using the assumption on the relaxation parameters, we have $|\hat{\mathsf{g}}_r(0)| < 1$ for $r \in [\![2, Q+1]\!]$. Using the assumption $\omega_1^{(0)} = 1$ (*i.e.* $\hat{\mathsf{w}}_1(\xi\Delta x) = 1 + O(|\xi\Delta x|)$), Proposition 10.2.2 provides $\hat{\mathsf{g}}^{[r]}(\xi\Delta x) = 1 + O(|\xi\Delta x|)$ for $r \in [\![1, Q]\!]$. Therefore
$$[\hat{\mathsf{g}}^{[Q]}(\xi\Delta x),\dots,\hat{\mathsf{g}}^{[1]}(\xi\Delta x),\hat{\mathsf{w}}_1(\xi\Delta x)] = [\hat{\mathsf{g}}_1(\xi\Delta x)^Q,\dots,\hat{\mathsf{g}}_1(\xi\Delta x),1] + O(|\xi\Delta x|),$$
which means that the amplification factors of the initialisation schemes are the eigenvector of $\hat{\mathbf{Q}}(\xi\Delta x)$ associated with $\hat{\mathsf{g}}_1(\xi\Delta x)$ at leading order. Back in (10.33), this gives that
$$\begin{aligned}\hat{\mathsf{g}}^{[n]}(\xi\Delta x) &= \boldsymbol{e}_1^{\mathsf{t}}\hat{\mathbf{V}}(\xi\Delta x)\mathrm{diag}(\hat{\mathsf{g}}_1(\xi\Delta x)^{n-Q},\dots,\hat{\mathsf{g}}_{Q+1}(\xi\Delta x)^{n-Q})(\boldsymbol{e}_1 + O(|\xi\Delta x|)) \\ &= \boldsymbol{e}_1^{\mathsf{t}}\hat{\mathbf{V}}(\xi\Delta x)\mathrm{diag}(\hat{\mathsf{g}}_1(\xi\Delta x)^{n-Q}(1 + O(|\xi\Delta x|)),\hat{\mathsf{g}}_2(\xi\Delta x)^{n-Q}O(|\xi\Delta x|),\dots,\hat{\mathsf{g}}_{Q+1}(\xi\Delta x)^{n-Q}O(|\xi\Delta x|)) \\ &= \hat{\mathsf{g}}_1(\xi\Delta x)^n(1 + O(|\xi\Delta x|)) + \hat{\mathsf{g}}_2(\xi\Delta x)^nO(|\xi\Delta x|) + \cdots + \hat{\mathsf{g}}_{Q+1}(\xi\Delta x)^nO(|\xi\Delta x|)),\end{aligned}$$
where it is important to observe that the $O(|\xi\Delta x|)$-terms are independent of n. Considering that $|\hat{\mathsf{g}}_r(0)| < 1$ for $r \in [\![2, Q+1]\!]$, we deduce that $\lim_{n\to+\infty}\hat{\mathsf{g}}_r(\xi\Delta x)^n = 0$ for $r \in [\![2, Q+1]\!]$. Of course, convergence can be slow for high orders in the formal series. This entails that we have
$$\hat{\mathsf{g}}^{[n]}(\xi\Delta x) = \hat{\mathsf{g}}_1(\xi\Delta x)^n(1 + \hat{\mathsf{r}}^{[n]}(\xi\Delta x)),$$
where the residual $\hat{\mathsf{r}}^{[n]}(\xi\Delta x) = O(|\xi\Delta x|)$ is such that it converges to a fixed formal series for $n \to +\infty$. The usual trick provides
$$\lim_{n\to+\infty}\partial_t = \frac{\lambda}{\Delta x}\lim_{n\to+\infty}\frac{1}{n}\log(\hat{\mathsf{g}}^{[n]}(\xi\Delta x)) = \frac{\lambda}{\Delta x}\Big(\log(\hat{\mathsf{g}}_1(\xi\Delta x)) + \lim_{n\to+\infty}\frac{1}{n}\log(1 + \hat{\mathsf{r}}^{[n]}(\xi\Delta x)\Big) = \frac{\lambda}{\Delta x}\log(\hat{\mathsf{g}}_1(\xi\Delta x)).$$
- Observe that thanks to the stability assumption, there can be only one relaxation parameter $s_{\tilde{\imath}} = 2$. Otherwise, there would be a multiple eigenvalue on the unit circle for $\xi\Delta x = 0$, contradicting the stability assumption whilst generating linear instabilities. Up to a rearrangement of the moments, we have $\tilde{\imath} = 2$. By the assumptions on w_1 and the initialisation schemes, we deduce that
$$[\hat{\mathsf{g}}^{[Q]}(\xi\Delta x),\dots,\hat{\mathsf{g}}^{[1]}(\xi\Delta x),\hat{\mathsf{w}}_1(\xi\Delta x)] = [\hat{\mathsf{g}}_1(\xi\Delta x)^Q,\dots,\hat{\mathsf{g}}_1(\xi\Delta x),1] + O(|\xi\Delta x|^{H+1}),$$

which means that the amplification factors of the initialisation schemes are the eigenvector of $\hat{\mathbf{Q}}(\xi\Delta x)$ associated with $\hat{\mathrm{g}}_1(\xi\Delta x)$ up to order $O(|\xi\Delta x|^{H+1})$. Into (10.33), this yields

$$\begin{aligned}\hat{\mathrm{g}}^{[n]}(\xi\Delta x) &= \boldsymbol{e}_1^{\mathrm{t}}\hat{\mathbf{V}}(\xi\Delta x)\mathrm{diag}(\hat{\mathrm{g}}_1(\xi\Delta x)^{n-Q},\hat{\mathrm{g}}_2(\xi\Delta x)^{n-Q},\dots,\hat{\mathrm{g}}_{Q+1}(\xi\Delta x)^{n-Q})(\boldsymbol{e}_1+O(|\xi\Delta x|^{H+1}))\\ &= \hat{\mathrm{g}}_1(\xi\Delta x)^n(1+O(|\xi\Delta x|^{H+1}))+\hat{\mathrm{g}}_2(\xi\Delta x)^nO(|\xi\Delta x|^{H+1})+\cdots+\hat{\mathrm{g}}_{Q+1}(\xi\Delta x)^nO(|\xi\Delta x|^{H+1})),\end{aligned}$$

where all the $O(|\xi\Delta x|^{H+1})$-terms are independent of n. Due to the fact that $\hat{\mathrm{g}}_2(0)=1-s_2=-1$, the formal series $\hat{\mathrm{g}}_2(\xi\Delta x)^n$ contains terms that can oscillate by featuring expressions involving $(-1)^n$, and the term at order $h\in[\![0,+\infty[\![$ grows with n at most as a polynomial of degree h in n. We indicate this fact using the notation $\hat{\mathrm{g}}_2(\xi\Delta x)^n=\sum_{h=0}^{h=+\infty}O(n^h)(\xi\Delta x)^h$. Therefore

$$\hat{\mathrm{g}}_2(\xi\Delta x)^nO(|\xi\Delta x|^{H+1})=\sum_{h=H+1}^{+\infty}O(n^{h-H-1})(\xi\Delta x)^h.$$

As previously acknowledged, since $|\hat{\mathrm{g}}_r(0)|<1$ for $r\in[\![3,Q+1]\!]$, we deduce that $\lim_{n\to+\infty}\hat{\mathrm{g}}_r(\xi\Delta x)^n=0$ for $r\in[\![3,Q+1]\!]$. This ensures that

$$\hat{\mathrm{g}}^{[n]}(\xi\Delta x)=\hat{\mathrm{g}}_1(\xi\Delta x)^n\Big(1+\sum_{h=H+1}^{+\infty}O(n^{h-H-1})(\xi\Delta x)^h\Big).$$

Utilising the usual trick, we have

$$\begin{aligned}\lim_{n\to+\infty}\partial_t &= \frac{\lambda}{\Delta x}\lim_{n\to+\infty}\frac{1}{n}\log(\hat{\mathrm{g}}^{[n]}(\xi\Delta x))=\frac{\lambda}{\Delta x}\Big(\log(\hat{\mathrm{g}}_1(\xi\Delta x))+\lim_{n\to+\infty}\frac{1}{n}\log\Big(1+\sum_{h=H+1}^{+\infty}O(n^{h-H-1})(\xi\Delta x)^h\Big)\Big)\\ &= \frac{\lambda}{\Delta x}\Big(\log(\hat{\mathrm{g}}_1(\xi\Delta x))+\lim_{n\to+\infty}\frac{1}{n}\sum_{h=H+1}^{+\infty}O(n^{h-H-1})(\xi\Delta x)^h\Big)=\frac{\lambda}{\Delta x}\Big(\log(\hat{\mathrm{g}}_1(\xi\Delta x))+\lim_{n\to+\infty}\sum_{h=H+1}^{+\infty}O(n^{h-H-2})(\xi\Delta x)^h\Big)\\ &= \frac{\lambda}{\Delta x}\Big(\log(\hat{\mathrm{g}}_1(\xi\Delta x))+\lim_{n\to+\infty}\Big(O(n^{-1})(\xi\Delta x)^{H+1}+\sum_{h=H+2}^{+\infty}O(n^{h-H-2})(\xi\Delta x)^h\Big)\Big)\\ &= \frac{\lambda}{\Delta x}\Big(\log(\hat{\mathrm{g}}_1(\xi\Delta x))+\lim_{n\to+\infty}\sum_{h=H+2}^{+\infty}O(n^{h-H-2})(\xi\Delta x)^h\Big),\end{aligned}$$

achieving the demonstration.

□

10.2.8 Conclusions

In Section 10.2, we have proposed a way of linking the initial datum of the lattice Boltzmann scheme $\mathbf{m}(0,\cdot)$ to the initial datum u° of the Cauchy problem (10.2). This allowed us to propose a modified equation analysis of the initialisation phase—see Proposition 10.2.1 and Proposition 10.2.2—making the study of the real behaviour of the numerical schemes possible and find the constraints—see Corollary 10.2.1 and Corollary 10.2.2—under which the initialisation schemes are consistent with the same equation (10.1) as the bulk scheme, preventing from having order reductions. We have also stressed that controlling the behaviour of the scheme inside the initialisation layer implies a control on the numerical scheme eventually in time (Proposition 10.2.3). The general computations have been done until order $O(\Delta x)$ but can be carried further to $O(\Delta x^2)$ and above for specific schemes, as in Section 10.3 and Section 10.4. This provides additional information on other features of the schemes close to the beginning of the simulation, such as dissipation and dispersion.

10.3 Examples and numerical simulations

Section 10.3 first aims at checking the previously introduced theory concerning consistency on actual numerical simulations on the $\mathrm{D_1Q_2}$ (*cf.* Example 10.1.1). Moreover, the computations of the modified equation shall be pushed

one order further providing the dissipation of the starting schemes, the impact of which is precisely quantified on the numerical experiments for a D_1Q_2 and D_1Q_3 scheme. Finally, the example of D_1Q_3 scheme paves the way for the general discussion of Section 10.4 concerning a more precise counting of the number of initialisation schemes—based on the observations in Section 7.6.3—with important consequences on the dissipation of the numerical schemes.

10.3.1 D_1Q_2 scheme

Consider the scheme from Example 10.1.1. The modified equation of the bulk Finite Difference scheme reads as in [Graille, 2014] and Theorem 10.2.1

$$\partial_t\phi(t,x)+\lambda\epsilon_2\partial_x\phi(t,x)-\lambda\Delta x\Big(\frac{1}{s_2}-\frac{1}{2}\Big)(1-\epsilon_2^2)\partial_{xx}\phi(t,x)=O(\Delta x^2),\qquad (t,x)\in\mathbb{R}_+\times\mathbb{R},\tag{10.34}$$

thus to be consistent with the Cauchy problem (10.1), one takes $\epsilon_2=V/\lambda$. For $s_2<2$, the bulk Finite Difference scheme is first-order accurate, thus initialisation schemes which are non-consistent with the target conservation law—*i.e.* indeed violating (10.17) or (10.24)—do not degrade the order of convergence. For $s_2=2$, the bulk Finite Difference scheme is second-order accurate, thus consistent initialisation schemes are needed, *i.e.* verifying (10.17) or (10.24). Observe that the scheme is L^2 stable according to the conditions of Theorem 7.7.3 are met) under the conditions ([Graille, 2014] and Appendix A.1)

$$s_2\in]0,2],\qquad\text{and}\qquad\begin{cases}|\epsilon_2|\leq 1, & \text{if}\quad s_2\in]0,2[,\\ |\epsilon_2|<1, & \text{if}\quad s_2=2.\end{cases}\tag{10.35}$$

The conditions delimited by brackets are the Courant-Friedrichs-Lewy (CFL) condition, which is known to be strict for the leap-frog scheme.

We consider five different choices of initialisation schemes. They are designed to showcase different facets of the previous theoretical discussion. More precisely, the first initialisation is the one where all data are taken at equilibrium, which is likely the most common way of initializing lattice Boltzmann schemes [Graille, 2014, Caetano et al., 2023]. The second and the third initialisations both render a forward centered scheme as initialisation scheme, which would be unstable if used as bulk scheme. Still, these two initialisations yield different outcomes for the associated numerical simulations and our theory accounts for this phenomenon. The fourth initialisation aims at obtaining a Lax-Wendroff initialisation scheme, which allows to study the effect of a second-order initialisation scheme. Finally, the fifth initialisation is inspired by works from the literature [Van Leemput et al., 2009].

- **Lax-Friedrichs scheme** (LF), a first-order consistent scheme which we shall obtain using the local initialisation

$$\mathsf{w}_1=1,\qquad \mathsf{w}_2=\epsilon_2.\tag{10.36}$$

  Except when $s_2=1$ (where $Q=0$), the dissipation of the bulk Finite Difference scheme is not matched by the one of the Lax-Friedrichs scheme.

- **Forward centered scheme** (FC). This is a first-order consistent scheme which is unstable even under CFL condition (10.35) if used as bulk scheme, due to its negative dissipation. Still, it is perfectly suitable for the initialisation of the method (see [Strikwerda, 2004, Chapter 10]). Its diffusivity shall not match the one of the bulk Finite Difference scheme, see (10.34). This initialisation scheme cannot stem from a local initialisation, *i.e.* $\mathsf{w}_1,\mathsf{w}_2\in\mathbb{R}$, since the only first-order consistent initialisation scheme that can be obtained in this way is the Lax-Friedrichs scheme (10.36). We could unsuccessfully try to generate it by a local initialisation of the conserved moment, that is $\mathsf{w}_1=1$ and prepared initialisation of the non-conserved one, thus $\mathsf{w}_2\in\mathsf{D}$. Considering—see Appendix A.2 for the details—a prepared initialisation for both moments, thus $\mathsf{w}_1,\mathsf{w}_2\in\mathsf{D}$, several choices are possible to recover this scheme. One is

$$w_{1,\pm1}=\frac{1}{2},\qquad w_{2,\pm1}=\mp\frac{1\pm s_2\epsilon_2}{2(1-s_2)},\qquad w_{2,0}=\frac{\epsilon_2}{1-s_2},\tag{10.37}$$

Table 10.1: Expected order of convergence in Δx for the D_1Q_2 scheme.

Test	Bulk scheme 1st order ($0 < s_2 < 2$)	Bulk scheme 2nd order ($s_2 = 2$)
(a) - (7.53)	order 1/4	order 1/3
(b) - (7.54)	order 3/4	order 1
(c) - (7.55)	order 1	order 5/3
(d) - (7.56)	order 1	order 2

and agrees with (10.22), (10.23) and (10.24). Another possible choice to obtain the desired scheme would be

$$w_{1,\pm2} = \pm\frac{\epsilon_2}{2}, \qquad w_{1,\pm1} = \frac{1}{2}, \qquad w_{2,\pm2} = -\frac{\epsilon_2(1 \pm s_2\epsilon_2)}{2(1-s_2)}, \qquad w_{2,\pm1} = \mp\frac{1 \pm s_2\epsilon_2}{2(1-s_2)}. \tag{10.38}$$

However, this initialisation yields only (10.22) but does not fulfill either (10.23) or (10.24). This means that in this case m_1 is initialized as a first-order perturbation of the datum of the Cauchy problem (10.2) and that m_2 is not initialized at equilibrium at leading order.

- **Lax-Wendroff scheme** (LW). This is a second-order consistent scheme with no dissipation, thus matches the diffusivity of the bulk Finite Difference scheme when $s_2 = 2$. Remark that since the bulk scheme is at most second-order accurate, it is somehow excessive to initialize with a scheme of the same order. Following an analogous procedure to the centered forward scheme, one possible initialisation is $w_1, w_2 \in \mathsf{D}$ with

$$w_{1,\pm1} = \frac{1-\epsilon_2^2}{2}, \quad w_{1,0} = \epsilon_2^2, \quad w_{2,\pm1} = \mp\frac{(1 \pm s_2\epsilon_2)(1-\epsilon_2^2)}{2(1-s_2)}, \quad w_{2,0} = \frac{\epsilon_2(1-s_2\epsilon_2^2)}{1-s_2}, \tag{10.39}$$

according to (10.21), which respects (10.22), (10.23) and (10.24). Again, it is also possible to generate initialisations yielding this scheme which do not fulfill (10.23) and (10.24).

- **Smooth initialisation inspired by** [Van Leemput et al., 2009] (RE1). The idea of this initialisation is to make the most of the terms in the modified equation of the initialisation schemes and that of the bulk Finite Difference scheme to match, if possible, without modification the conserved moment, that is $w_1 \in \mathbb{R}$. In particular, in our case, this allows to match the numerical diffusion coefficient between the two schemes for every $s_2 \in]0,2]$, as we shall see. We adapt Equation (13) from [Van Leemput et al., 2009] by discretising the continuous derivative by a second-order centered formula, having

$$w_1 = 1 \quad \text{and} \quad w_2 \in \mathsf{D}, \quad \text{where} \quad w_{2,\pm1} = \pm\frac{1-\epsilon_2^2}{2s_2}, \qquad w_{2,0} = \epsilon_2, \tag{10.40}$$

according to (10.21). This initialisation fulfills (10.22), (10.23) and (10.24).

10.3.1.1 Study of the convergence order

To empirically analyze the preservation of the order of the bulk Finite Difference scheme, we consider the following initial data with different smoothness, namely (7.53), (7.54), (7.55) and (7.56) that we have already used in Section 7.7.3. As common in the linear framework, we monitor the ℓ^2 errors. We simulate for $\lambda = 1$, $\epsilon_2 = V/\lambda = 1/2$ with final time $T = 1/2$ and on a bounded domain $\Omega = [-1, 1]$ with periodic boundary conditions.

We expect the scheme to be convergent following the orders given in Table 10.1 [Strikwerda, 2004, Chapter 10] and Section 7.7.3 and observe orders exceeding one provided that both following conditions are met:

1. the initialisation scheme is at least first-order consistent with the Cauchy problem (10.1);
2. the initial filter on the initial datum w_1 is such that $\omega_1^{(1)} = 0$, meaning that it perturbs from $O(\Delta x^2)$ or for higher orders.

The results are in agreement with the theory. We just present few of them for the sake of avoiding redundancy, in particular, those concerning the forward centered initialisation schemes (10.37) and (10.38) given in Figure 10.3. As expected, despite the fact that the obtained initialisation scheme is the same, (10.38) pollutes the initial datum

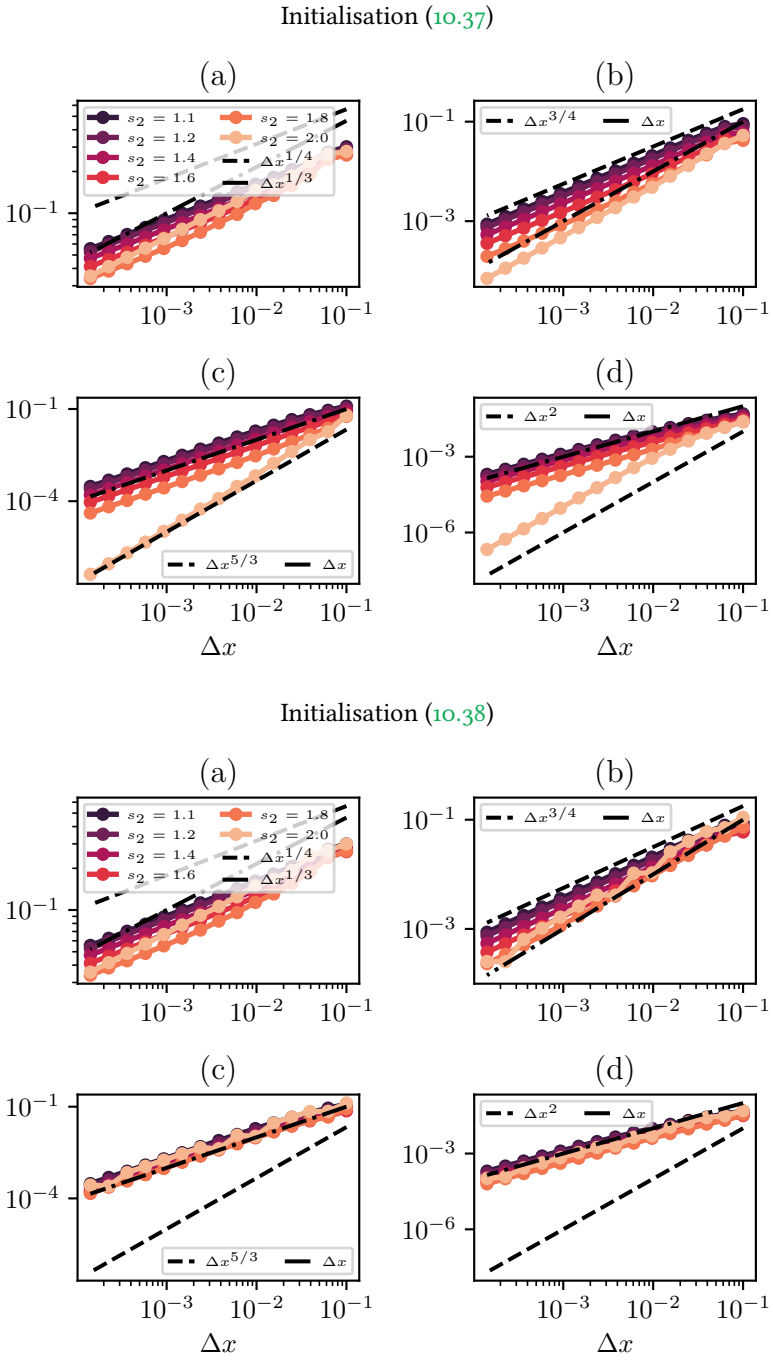


Figure 10.3: ℓ^2 errors at the final time T for two forward centered initialisations (10.37) (top) and (10.38) (bottom). Since the letter irremediably perturbs the conserved moment feeding the bulk Finite Difference scheme, the orders of convergence above one are lowered.

with respect to the one from the Cauchy problem (10.2) due to a first-order term $\omega_1^{(1)} \neq 0$. Hence, even for $s_2 = 2$, the order of convergence is lowered. We shall reinterpret why (10.38) yields a poor behaviour.

10.3.1.2 Study of the time smoothness of the numerical solution

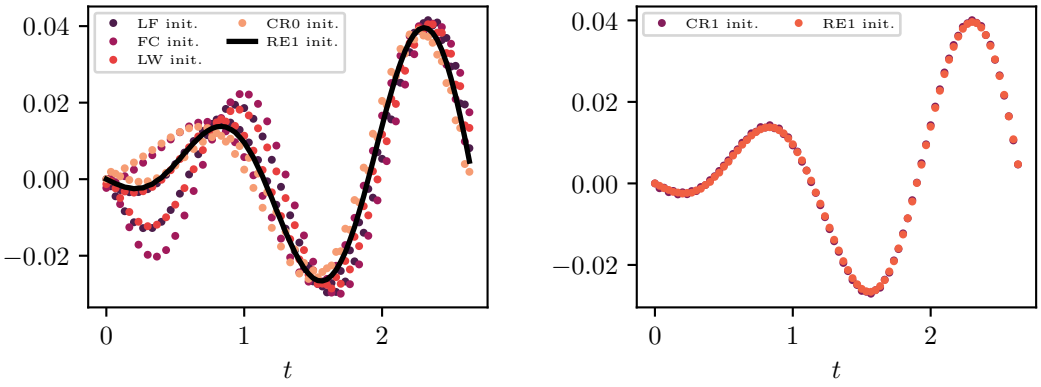


Figure 10.4: Test for the smoothness in time close to $t = 0$ for $s_2 = 1.99$: difference between exact and numerical solution at the eighth lattice point.

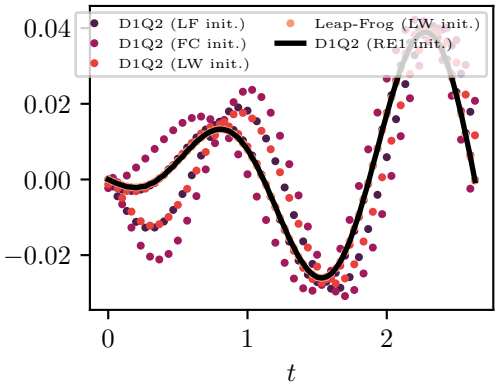


Figure 10.5: Test for the smoothness in time close to $t = 0$ for $s_2 = 2$: difference between exact and numerical solution at the eighth lattice point. Compared to Figure 10.4, CR0 and CR1 from [Van Leemput et al., 2009] cannot be used.

We have observed that the only proposed initialisation matching the dissipation of the bulk scheme for every $s_2 \in]0, 2]$ is the one given by (10.40). To confirm that this is the origin of its good performances in term of time smoothness of the discrete solution close to the initial time, we repeat the numerical experiment found in [Van Leemput et al., 2009]. The simulation is carried on the periodic domain $\Omega = [0, 1]$ discretized with $\Delta x = 1/30$, $s_2 = 1.99$, $\lambda = 1$ and $\epsilon_2 = V/\lambda = 0.66$. The initial datum of the Cauchy problem is $u^\circ(x) = \cos(2\pi x)$.

We initialize using the Lax-Friedrichs initialisation (10.36) (coinciding with what [Van Leemput et al., 2009] calls RE0 scheme), the forward centered initialisation (10.37), the Lax-Wendroff initialisation (10.39), the RE1 initialisation (10.40) and the implicit initialisations CR0 and CR1 proposed in [Van Leemput et al., 2009], which are not detailed here. We measure the difference between the exact solution and the approximate solution at the eighth cell of the lattice. The results are given in Figure 10.4 and are in accordance with the previous analysis as well as the computations in [Van Leemput et al., 2009]. Indeed, since the dissipation of the bulk Finite Difference scheme is almost zero for $s_2 = 1.99$, the Lax-Wendroff scheme is supposed to almost match this dissipation. However here, the same phenomenon that took place in Section 10.3.1.1 at leading order for the forward centered initialisation

between (10.37) and (10.38), due to the introduction of a first-order perturbation on the conserved moment, now takes place for (10.39), because it introduces a second-order perturbation on the conserved moment m_1 feeding the multi-step bulk Finite Difference scheme (10.9), namely $\omega_1^{(2)} \neq 0$. Taking $s_2 = 2$, hence no dissipation from the bulk scheme, we obtain the result in Figure 10.5, which is not different from the previous one (notice that here the implicit initialisation RE1 cannot be utilized). In this Figure, we have also repeated the simulation using a leap-frog scheme (coinciding with the bulk Finite Difference scheme) initialized with a Lax-Wendroff scheme, which conversely leads the expected smoothness, since we do not have to filter the initial datum of the Cauchy problem.

10.3.1.3 Theoretical analysis using the modified equations

Let us proceed to a more quantitative study of what can be observed in Figure 10.4 and Figure 10.5. To this end, we push the computation of the modified equation of the starting schemes for $n \in \mathbb{N}^*$ to order $O(\Delta x^2)$. To complete the previous computations, we are left to consider

$$\begin{aligned}(\boldsymbol{\mathcal{E}}^n)^{(2)} &= \sum\{\text{per. of } \boldsymbol{\mathcal{E}}^{(0)} \ (n-1 \text{ tm.}) \text{ and } \boldsymbol{\mathcal{E}}^{(2)} \ (\text{once})\} + \sum\{\text{per. of } \boldsymbol{\mathcal{E}}^{(0)} \ (n-2 \text{ tm.}) \text{ and } \boldsymbol{\mathcal{E}}^{(1)} \ (\text{twice})\} \\ &= \sum_{r=0}^{r=n-1} (\boldsymbol{\mathcal{E}}^{(0)})^r \boldsymbol{\mathcal{E}}^{(2)} (\boldsymbol{\mathcal{E}}^{(0)})^{n-1-r} + \sum_{r=0}^{r=n-2} \sum_{p=0}^{p=n-2-r} (\boldsymbol{\mathcal{E}}^{(0)})^r \boldsymbol{\mathcal{E}}^{(1)} (\boldsymbol{\mathcal{E}}^{(0)})^p \boldsymbol{\mathcal{E}}^{(1)} (\boldsymbol{\mathcal{E}}^{(0)})^{n-2-r-p},\end{aligned}$$

where $\boldsymbol{\mathcal{E}} \asymp \mathbf{E}$ with $\mathbf{E}$ being the scheme matrix. Using the matrix from the particular $\mathrm{D_1Q_2}$ scheme, we obtain for every $n \in \mathbb{N}^*$

$$\begin{aligned}(\boldsymbol{\mathcal{E}}^n)_{11}^{(2)} = \Big(&\frac{n}{2} + \sum_{r=0}^{r=n-2} \sum_{p=1}^{p=n-1-r} (1-s_2)^p \\ &+ \epsilon_2^2 \sum_{r=0}^{r=n-2} \sum_{p=0}^{p=n-2-r} \Big(s_2^2 + s_2(1-s_2)\pi_{n-2-r-p}(s_2) + (1-s_2)\pi_p(s_2)\pi_{n-1-r-p}(s_2)\Big)\Big)\partial_{xx},\end{aligned}$$

and

$$(\boldsymbol{\mathcal{E}}^n)_{12}^{(2)} = \epsilon_2 \sum_{r=0}^{n-2} \sum_{p=0}^{n-2-r} (1-s_2)^{n-1-r-p} \pi_{p+1}(s_2) \partial_{xx},$$

where we recall that $\pi_r(X) = 1-(1-X)^r$ for $r \in \mathbb{N}$. With the usual procedure, we obtain for $n \in \mathbb{N}^*$ and $x \in \mathbb{R}$

$$\begin{aligned}&\partial_t \phi(0,x) - \frac{\lambda}{n}\Big((\boldsymbol{\mathcal{E}}^n)_{11}^{(1)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(1)} \omega_2^{(0)} + \omega_1^{(1)}\Big)\phi(0,x) \\ &+ n\frac{\Delta x}{2\lambda}\partial_{tt}\phi(0,x) - \frac{\lambda \Delta x}{n}\Big((\boldsymbol{\mathcal{E}}^n)_{11}^{(2)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(2)}\omega_2^{(0)} + (\boldsymbol{\mathcal{E}}^n)_{11}^{(1)}\omega_1^{(1)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(1)}\omega_2^{(1)} + \omega_1^{(2)}\Big)\phi(0,x) = O(\Delta x^2).\end{aligned}$$

- **Lax-Friedrichs** (10.36).

Proposition 10.3.1: Modified equations for the starting schemes under (10.36)

Under acoustic scaling, the modified equations for the starting schemes for the Lax-Friedrichs initialisation given by (10.36) are, for $n \in \mathbb{N}^*$

$$\partial_t \phi(0,x) + \lambda\epsilon_2 \partial_x \phi(0,x) - \lambda\Delta x\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1 - \frac{r}{n}\Big)(1-s_2)^r\Big)(1-\epsilon_2^2)\partial_{xx}\phi(0,x) = O(\Delta x^2), \qquad x \in \mathbb{R}. \quad (10.41)$$

Proof. This initialisation fulfils the requirements by Corollary 10.2.2, which leads to

$$\partial_t \phi(0,x) + \lambda\epsilon_2 \partial_x \phi(0,x) + n\frac{\Delta x}{2\lambda}\partial_{tt}\phi(0,x) - \frac{\lambda\Delta x}{n}\Big((\boldsymbol{\mathcal{E}}^n)_{11}^{(2)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(2)}\epsilon_2\Big)\phi(0,x) = O(\Delta x^2), \quad (10.42)$$

for $n \in \mathbb{N}^*$ and $x \in \mathbb{R}$. Using the previous order to get rid of the second-order time derivative ∂_{tt} [Warming and Hyett, 1974, Carpentier et al., 1997, Dubois, 2008, Dubois, 2022] boils down to

$$\partial_t \phi(0,x) + \lambda\epsilon_2 \partial_x \phi(0,x) - \lambda\Delta x\Big(-\frac{n}{2}\epsilon_2^2\partial_{xx} + \frac{1}{n}\Big((\boldsymbol{\mathcal{E}}^n)_{11}^{(2)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(2)}\epsilon_2\Big)\Big)\phi(0,x) = O(\Delta x^2),$$

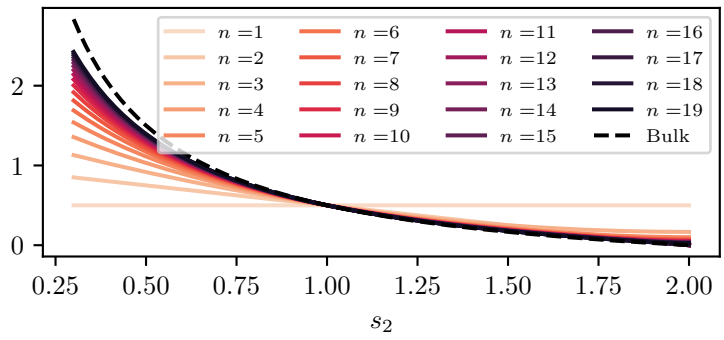


Figure 10.6: Plot of the polynomial $1/2+\sum_{r=1}^{r=n-1}(1-r/n)(1-s_2)^r$ appearing in (10.41) for different n compared to $1/s_2-1/2$ (bulk).

for $n\in\mathbb{N}^*$ and $x\in\mathbb{R}$. We are left to deal with the diffusion term, for $n\in\mathbb{N}^*$

$$(\boldsymbol{\mathcal{E}}^n)^{(2)}_{11}+(\boldsymbol{\mathcal{E}}^n)^{(2)}_{12}\epsilon_2=\Big(\frac{n}{2}+\sum_{r=0}^{n-2}\sum_{p=1}^{n-1-r}(1-s_2)^p+\epsilon_2^2\sum_{r=0}^{n-2}\sum_{p=0}^{n-2-r}\Big(s_2^2+s_2(1-s_2)\pi_{n-2-r-p}(s_2)$$
$$+(1-s_2)\pi_p(s_2)\pi_{n-1-r-p}(s_2)+(1-s_2)^{n-1-r-p}\pi_{p+1}(s_2)\Big)\Big)\partial_{xx}.$$

Using the expression for π_r to handle the last term shows that

$$(\boldsymbol{\mathcal{E}}^n)^{(2)}_{11}+(\boldsymbol{\mathcal{E}}^n)^{(2)}_{12}\epsilon_2=\Big(\frac{n}{2}+\sum_{r=1}^{n-1}(n-r)(1-s_2)^r+\epsilon_2^2\Big(\frac{n(n-1)}{2}-\sum_{r=1}^{n-1}(n-r)(1-s_2)^r\Big)\Big)\partial_{xx},$$

for $n\in\mathbb{N}^*$. Plugging into the expansion (10.42) provides

$$\partial_t\phi(0,x)+\lambda\epsilon_2\partial_x\phi(0,x)-\lambda\Delta x\Big(\frac{1}{2}+\sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r\Big)(1-\epsilon_2^2)\partial_{xx}\phi(0,x)=O(\Delta x^2),\qquad n\in\mathbb{N}^*,$$

for $x\in\mathbb{R}$. □

This proves once more that the origin of the initial boundary layer is the mismatch in the dissipation coefficient of the scheme, see Figure 10.6. Of course, it must be kept in mind that these expansions are meaningful as long as $n\Delta t\ll 1$, this is, for small times. However, from the simulations and Figure 10.6, we see that the boundary layer damps in time, since the dissipation coefficient in (10.41) converges to the bulk one in (10.34) by taking the formal limit $n\to+\infty$:

$$\lim_{n\to+\infty}\Big(\frac{1}{2}+\sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r\Big)=\lim_{n\to+\infty}\Big(\frac{1}{2}+\frac{(1-s_2)^{n+1}}{ns_2^2}-\frac{(1-s_2)}{ns_2^2}+\frac{(1-s_2)}{s_2}\Big)=\frac{1}{s_2}-\frac{1}{2},$$

unsurprisingly by virtue of Proposition 10.2.4. We see that—as previously claimed—this formal limit holds regardless of the fulfilment of the CFL condition. However, it strongly depends on the fact that $s_2\le 2$, otherwise, it would not hold and it would exponentially diverge. We can also study the behaviour for $s_2\simeq 2$:

$$\lim_{s_2\to 2^-}\frac{1}{2}+\sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r=\frac{1}{2}+\sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(-1)^r=\frac{1-(-1)^n}{4n}=\begin{cases}0, & \text{for } n \text{ even},\\ 1/(2n), & \text{for } n \text{ odd},\end{cases}$$

for $n\in\mathbb{N}^*$. This explains why the errors in Figure 10.4 and Figure 10.5 are close to the ones of RE1 (10.40) (up to high order contributions) for even time steps. On the one hand, for n even, the dissipation of the bulk Finite Difference scheme is matched by the starting schemes, producing good agreement. On the other hand, for n odd, the dissipation is strictly positive, though decreasing linearly with n, creating the jumping behaviour of the errors. This suggests that the damping of the initial boundary layer should be proportional

to t^{-1} and explains the discrepancies with respect to RE1 (10.40) for the odd time steps. Finally, observe that this decoupling—even as far as the dissipation is concerned—between even and odd time steps for $s_2 = 2$ is expected since the bulk Finite Difference scheme is a leap-frog.

- **Forward centered scheme** (10.37).

Proposition 10.3.2: Modified equations for the starting schemes under (10.37)

Under acoustic scaling, the modified equations for the starting schemes for the forward centered initilisation given by (10.37) are, for $n \in \mathbb{N}^*$

$$\begin{aligned}&\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) + O(\Delta x^2) \\ &\quad - \lambda\Delta x\Big(\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r\Big)(1-\epsilon_2^2) + \frac{1}{2n}\Big(1-2\sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)\partial_{xx}\phi(0,x) = 0,\end{aligned} \tag{10.43}$$

with $x \in \mathbb{R}$.

Proof. This scheme fulfills the conditions of Corollary 10.2.2, hence for $n \in \mathbb{N}^*$

$$\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) - \lambda\Delta x\Big(-\frac{n}{2}\epsilon_2^2\partial_{xx} + \frac{1}{n}\Big((\boldsymbol{\mathcal{E}}^n)_{11}^{(2)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(2)}\epsilon_2 + (\boldsymbol{\mathcal{E}}^n)_{12}^{(1)}\omega_2^{(1)} + \omega_1^{(2)}\Big)\Big)\phi(0,x) = O(\Delta x^2),$$

for $x \in \mathbb{R}$, where only the terms $\omega_2^{(1)} = 1/(1-s_2)\partial_x$ and $\omega_1^{(2)} = 1/2\partial_{xx}$ introduce discrepancies from the Lax-Friedrichs initialisation (10.36). Using (10.44) we obtain for $n \in \mathbb{N}^*$ and $x \in \mathbb{R}$

$$\begin{aligned}&\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) \\ &\quad - \lambda\Delta x\Big(\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r\Big)(1-\epsilon_2^2) + \frac{1}{2n}\Big(1-2\sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)\partial_{xx}\phi(0,x) = O(\Delta x^2).\end{aligned}$$

□

Again, according to Proposition 10.2.4, the bulk viscosity coefficient is asymptotically reached, since

$$\lim_{n\to+\infty}\Big(\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r\Big)(1-\epsilon_2^2) + \frac{1}{2n}\Big(1-2\sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big) = \Big(\frac{1}{s_2} - \frac{1}{2}\Big)(1-\epsilon_2^2).$$

Concerning the behaviour close to $s_2 \simeq 2$, we have

$$\begin{aligned}&\lim_{s_2\to 2^-}\Big(\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1-\frac{r}{n}\Big)(1-s_2)^r\Big)(1-\epsilon_2^2) + \frac{1}{2n}\Big(1-2\sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big) \\ &= \frac{(1-(-1)^n)}{4n}(1-\epsilon_2^2) + \frac{(-1)^n}{2n} = \begin{cases} 1/(2n), & \text{for } n \text{ even}, \\ -\epsilon_2^2/(2n), & \text{for } n \text{ odd}, \end{cases}\end{aligned}$$

for $n \in \mathbb{N}^*$. We observe that the even steps of starting schemes have the same diffusivity as the odd steps for the Lax-Friedrichs initialisation (10.36), whereas the odd ones have negative diffusivity, which remains from having an initialisation scheme with negative dissipation, coupled with the fact that the bulk Finite Difference scheme is a leap-frog scheme. The question which might be risen is on how the overall scheme can remain stable. In terms of Finite Differences, the choice of initial datum only changes the spectrum of the data feeding the bulk Finite Difference scheme, which is stable under (10.35), for every initial datum. Concerning the previous computation, we have that under the CFL condition $-\epsilon_2^2/(2n) \geq -1/(2n)$, hence steps with negative dissipation are compensated by steps with sufficiently positive dissipation, yielding an overall stable scheme.

- **Forward centered scheme** (10.38). For this scheme, it is useless to analyze until second order because we know that issues start at $O(\Delta x)$, see Section 10.3.1.1.

Proposition 10.3.3: Modified equations for the starting schemes under (10.38)

Under acoustic scaling, the modified equations for the starting schemes for the forward centered initilisation given by (10.38) are, for $n \in \mathbb{N}^*$

$$\partial_t \phi(0,x) + \lambda\epsilon_2\Big(1 + \frac{2}{n}\Big(1 - \sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)\partial_x\phi(0,x) = O(\Delta x), \qquad x \in \mathbb{R}.$$

Proof. We have

$$\partial_t \phi(0,x) - \frac{\lambda}{n}\Big((\mathcal{E}^n)_{11}^{(1)} + (\mathcal{E}^n)_{12}^{(1)}\omega_2^{(0)} + \omega_1^{(1)}\Big)\phi(0,x) = O(\Delta x), \qquad n \in \mathbb{N}^*, \quad x \in \mathbb{R}.$$

where in this case $\omega_2^{(0)} = -(1+s_2)/(1-s_2)\epsilon_2$ and $\omega_1^{(1)} = -2\epsilon_2\partial_x$. Recalling that

$$(\mathcal{E}^n)_{11}^{(1)} = -\epsilon_2 \sum_{r=0}^{n-1} \pi_{n-r}(s_2)\partial_x, \qquad (\mathcal{E}^n)_{12}^{(1)} = -\sum_{r=0}^{n-1}(1-s_2)^{n-r}\partial_x, \qquad n \in \mathbb{N}^*, \tag{10.44}$$

yields

$$(\mathcal{E}^n)_{11}^{(1)} + (\mathcal{E}^n)_{12}^{(1)}\omega_2^{(0)} + \omega_1^{(1)} = -\epsilon_2\Big(n + 2\Big(1 - \sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)\partial_x, \qquad n \in \mathbb{N}^*,$$

thus

$$\partial_t \phi(0,x) + \lambda\epsilon_2\Big(1 + \frac{2}{n}\Big(1 - \sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)\partial_x\phi(0,x) = O(\Delta x), \qquad n \in \mathbb{N}^*, \quad x \in \mathbb{R}.$$

□

Unsurprisingly, the initialisation scheme is consistent ($n = 1$), but the general starting schemes ($n > 1$) are not. This does not prevent the overall scheme to converge, since $\omega_1^{(0)} = 1$ but only at first-order even when $s_2 = 2$, see Figure 10.3. Following Proposition 10.2.4

$$\lim_{n\to+\infty}\Big(1 + \frac{2}{n}\Big(1 - \sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big) = 1.$$

- **Lax-Wendroff** (10.39).

Proposition 10.3.4: Modified equations for the starting schemes under (10.39)

Under acoustic scaling, the modified equations for the starting schemes for the Lax-Wendroff initilisation given by (10.39) are, for $n \in \mathbb{N}^*$

$$\begin{aligned}&\partial_t \phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) + O(\Delta x^2) \\ &\quad - \lambda\Delta x\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1 - \frac{r}{n}\Big)(1-s_2)^r + \frac{1}{2n}\Big(1 - 2\sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)(1-\epsilon_2^2)\partial_{xx}\phi(0,x) = 0,\end{aligned} \tag{10.45}$$

for $x \in \mathbb{R}$.

Proof. The computation is similar to the previous ones, taking into account that the only terms to change are $\omega_2^{(1)} = (1-\epsilon_2^2)/(1-s_2)\partial_x$ and $\omega_1^{(2)} = (1-\epsilon_2^2)/2\partial_{xx}$. This provides the modified equations. □

As expected, the dissipation coefficients tend to the one of the bulk scheme for $n \to +\infty$ and for $s_2 \simeq 2$, we find

$$\lim_{s_2\to 2^-}\Big(\frac{1}{2} + \sum_{r=1}^{n-1}\Big(1 - \frac{r}{n}\Big)(1-s_2)^r + \frac{1}{2n}\Big(1 - 2\sum_{r=0}^{n-1}(1-s_2)^r\Big)\Big)(1-\epsilon_2^2) = \frac{1+(-1)^n}{4n} = \begin{cases} 1/(2n), & \text{for } n \text{ even}, \\ 0, & \text{for } n \text{ odd}, \end{cases}$$

for $n \in \mathbb{N}^*$. This is the opposite situation compared to the Lax-Friedrichs initialisation (10.36) and again justifies the jumping behaviour compared to RE1 (10.40), see Figure 10.4 and Figure 10.5. Moreover, we further understand why we still observe the boundary layer: even if the initialisation scheme matches the zero diffusivity of the bulk scheme, the second-order modification $\omega_1^{(2)} \neq 0$ we have imposed on the initial datum to obtain such initialisation scheme reverberates over the following (even) time steps.

- **Smooth initialisation RE1** (10.40).

Proposition 10.3.5: Modified equations for the starting schemes under (10.40)

Under acoustic scaling, the modified equations for the starting schemes for the smooth initilisation RE1 given by (10.40) are, for $n \in \mathbb{N}^*$

$$\partial_t \phi(0,x) + \lambda \epsilon_2 \partial_x \phi(0,x) - \lambda \Delta x \Big(\frac{1}{s_2} - \frac{1}{2} \Big)(1-\epsilon_2^2)\partial_{xx}\phi(0,x) = O(\Delta x^2), \qquad x \in \mathbb{R}.$$

Proof. This scheme fulfills Corollary 10.2.2 and we have for $n \in \mathbb{N}^*$ and $x \in \mathbb{R}$

$$\partial_t \phi(0,x) + \lambda \epsilon_2 \partial_x \phi(0,x) - \lambda \Delta x \Big(-\frac{n}{2}\epsilon_2^2 \partial_{xx} + \frac{1}{n}\Big((\boldsymbol{\mathcal{E}}^n)_{11}^{(2)} + (\boldsymbol{\mathcal{E}}^n)_{12}^{(2)}\epsilon_2 + (\boldsymbol{\mathcal{E}}^n)_{12}^{(1)}\omega_2^{(1)} \Big) \Big) \phi(0,x) = O(\Delta x^2),$$

where only $\omega_2^{(1)} = -(1-\epsilon_2^2)/s_2 \partial_x$ introduces differences compared to the Lax-Friedrichs initialisation (10.36). We therefore obtain for $n \in \mathbb{N}^*$ and $x \in \mathbb{R}$

$$\partial_t \phi(0,x) + \epsilon_2 \partial_x \phi(0,x) - \lambda \Delta x \Big(\frac{1}{2} - \sum_{r=1}^{n-1} \Big(1 - \frac{r}{n}\Big)(1-s_2)^r + \frac{1}{n s_2} \sum_{r=1}^{n} (1-s_2)^r \Big)(1-\epsilon_2^2)\partial_{xx}\phi(0,x) = O(\Delta x^2). \quad (10.46)$$

One can easily show by induction that

$$\frac{1}{2} - \sum_{r=1}^{n-1} \Big(1 - \frac{r}{n}\Big)(1-s_2)^r + \frac{1}{n s_2} \sum_{r=1}^{n} (1-s_2)^r = \frac{1}{s_2} - \frac{1}{2}, \qquad n \in \mathbb{N}^*,$$

yielding the same modified equation as the bulk Finite Difference scheme. □

This explains, once more, the smooth behaviour observed in Figure 10.4 and Figure 10.5 and also shows an actual application of Proposition 10.2.3 for $H = 2$.

10.3.2 D_1Q_3 scheme

The previous case of D_1Q_2 scheme suggests that particular care must be adopted when prepared initialisations for the conserved moment m_1 are used (*i.e.* $w_1 \in D$). Therefore, in what follows, we treat only local initialisations for any moment. We are now interested in equating the dissipation of the initialisation schemes with the one of the bulk scheme for a richer scheme: the D_1Q_3. In particular, we look for a full characterisation of the conditions under which $w_1, w_2, w_3 \in \mathbb{R}$ yield initialisation schemes with the same dissipation as the bulk Finite Difference scheme.

10.3.2.1 Description of the scheme

We consider the D_1Q_3 scheme in Section 1.5.2 with moment matrix $\boldsymbol{M}$ given by (1.6) in dimensionless form. This provides

$$\mathsf{T} = \begin{bmatrix} \frac{1}{3}(2S(x_1)+1) & A(x_1) & \frac{1}{3}(S(x_1)-1) \\ \frac{2}{3}A(x_1) & S(x_1) & \frac{1}{3}A(x_1) \\ \frac{2}{3}(S(x_1)-1) & A(x_1) & \frac{1}{3}(S(x_1)+2) \end{bmatrix}, \qquad \boldsymbol{K} = \begin{bmatrix} 1 & 0 & 0 \\ s_2\epsilon_2 & 1-s_2 & 0 \\ s_3\epsilon_3 & 0 & 1-s_3 \end{bmatrix}.$$

The modified equation of the bulk Finite Difference scheme from Theorem 10.2.1 is

$$\partial_t \phi(t,x) + \lambda \epsilon_2 \partial_x \phi(t,x) - \lambda \Delta x \Big(\frac{1}{s_2} - \frac{1}{2} \Big) \Big(\frac{2}{3} - \epsilon_2^2 + \frac{\epsilon_3}{3} \Big) \partial_{xx}\phi(t,x) = O(\Delta x^2), \qquad (t,x) \in \mathbb{R}_+ \times \mathbb{R}. \quad (10.47)$$

To have a stable bulk method in the L^2 metric, the dissipation coefficient must not be negative, hence $\epsilon_3 < -2+3\epsilon_2^2$ is forbidden, because the modulus of the "consistency" (or "physical") eigenvalue would initially increase above one for small wave-numbers, causing the bulk Finite Difference scheme to be unstable. Sufficient conditions are more involved to determine but can be checked numerically, *cf.* Section 7.7.3. Observe that the (10.47) does not depend on the choice of s_3. To obtain consistency with (10.1), we have to enforce $\epsilon_2 = V/\lambda$. Furthermore, two leverages are available to make the bulk Finite Difference scheme second-order consistent with the (10.1), namely taking $s_2 = 2$ or $s_2 \in]0,2]$ and $\epsilon_3 = -2+3\epsilon_2^2$.

10.3.2.2 Conditions to achieve time smoothness of the numerical solution

Assuming that $s_2, s_3 \neq 1$, we have that $Q = 2$, thus two initialisation schemes are to consider. Their modified equations, computed with the previous techniques and considering local initialisations following the conditions by Corollary 10.2.1 - *i.e.* $w_1 = 1$ and $w_2 = \epsilon_2$ are as follows.

- **First initialisation scheme**: (10.8) for $n = 1$

$$\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) - \lambda\Delta x\Big(\frac{1}{3} - \frac{\epsilon_2^2}{2} + \frac{s_3\epsilon_3}{6} + \frac{(1-s_3)w_3}{6}\Big)\partial_{xx}\phi(0,x) = O(\Delta x^2), \qquad x \in \mathbb{R}. \tag{10.48}$$

  This scheme makes sense as initialisation scheme unless both $s_2 = s_3 = 1$ (*i.e.* $Q = 0$), where we observe that the diffusion coefficient in (10.48) becomes equal to the one from (10.47). In this case, the choice of w_3 is unimportant, as expected.

- **Second initialisation scheme**: (10.8) for $n = 2$

$$\begin{aligned}&\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x)\\ &- \lambda\Delta x\Big(\frac{(2-s_2)}{3} + \frac{(s_2-2)\epsilon_2^2}{2} + \frac{s_3(5-2s_2-s_3)\epsilon_3}{12} + \frac{(1-s_3)(4-2s_2-s_3)w_3}{12}\Big)\partial_{xx}\phi(0,x) = O(\Delta x^2),\end{aligned} \tag{10.49}$$

  for $x \in \mathbb{R}$. In the case where both $s_2 = s_3 = 1$ ($Q = 0$), we have the previously described situation. Taking $s_2 \neq 1$ and $s_3 = 1$ ($Q = 1$), we obtain the modified equation of the first starting scheme which is not an initialisation scheme

$$\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) - \lambda\Delta x s_2\Big(\frac{1}{s_2} - \frac{1}{2}\Big)\Big(\frac{2}{3} - \epsilon_2^2 + \frac{\epsilon_3}{3}\Big)\partial_{xx}\phi(0,x) = O(\Delta x^2), \qquad x \in \mathbb{R},$$

  which equals (10.47) up to the multiplication of the diffusion coefficient by s_2. This discrepancy is the remaining contribution of the initialisation on the evolution of the solution, as we have already observed for the D_1Q_2 in Section 10.3.1 for all initialisation except (10.40). Taking $s_2 = 1$ and $s_3 \neq 1$ ($Q = 1$), we have

$$\partial_t\phi(0,x) + \lambda\epsilon_2\partial_x\phi(0,x) - \lambda\Delta x\Big(\frac{1}{3} - \frac{\epsilon_2^2}{2} + \frac{s_3(3-s_3)\epsilon_3}{12} + \frac{(1-s_3)(2-s_3)w_3}{12}\Big)\partial_{xx}\phi(0,x) = O(\Delta x^2),$$

  for $x \in \mathbb{R}$, which is utterly different from (10.47): the choice of initialisation w_3 and the relaxation parameter s_3 influence the diffusivity, contrarily to (10.47).

Remark 10.3.1. *The previous discussion again confirms that, for starting schemes which are not initialisation schemes, the choice of initialisations and relaxation parameters can change the modified equations compared to the bulk Finite Difference scheme and thus the dynamics of the method close to the beginning of the simulation. Moreover, even some parameters that do not influence the modified equation of the bulk Finite Difference scheme at a given order (see s_3 in this example) impact the modified equations of the starting schemes.*

According to Proposition 10.2.3, it is enough to study the order $O(\Delta x^2)$ for the initialisation schemes to deduce the modified equations for any starting scheme. In order to match the diffusivity in both initialisation scheme, we

Table 10.2: Different choices of parameters for the D_1Q_3 scheme ensuring match at order $O(\Delta x^2)$ between initialisation schemes and bulk Finite Difference scheme.

Factors controlling dissipation		Leverages to obtain compatible dissipation	
$s_2 = 1$	$\epsilon_3 \geq -2+3\epsilon_2^2$	$s_3 = 1$, any w_3	(a)
		$s_3 \neq 1$, $w_3 = \epsilon_3$	(b)
$s_2 \neq 1$	$\epsilon_3 > -2+3\epsilon_2^2$	$s_3 = 2-s_2$, $w_3 = (2(-2+3\epsilon_2^2)+(s_2-2)\epsilon_3)/s_2$	(c)
	$\epsilon_3 = -2+3\epsilon_2^2$	$s_3 = 1$, any w_3	(d)
		$s_3 \neq 1$, $w_3 = \epsilon_3$	(e)

set the following system

$$\begin{cases} \frac{1}{3} - \frac{\epsilon_2^2}{2} + \frac{s_3\epsilon_3}{6} + \frac{(1-s_3)w_3}{6} & = \left(\frac{1}{s_2} - \frac{1}{2}\right)\left(\frac{2}{3} - \epsilon_2^2 + \frac{\epsilon_3}{3}\right), \\ \frac{(2-s_2)}{3} + \frac{(s_2-2)\epsilon_2^2}{2} + \frac{s_3(5-2s_2-s_3)\epsilon_3}{12} + \frac{(1-s_3)(4-2s_2-s_3)w_3}{12} & = \left(\frac{1}{s_2} - \frac{1}{2}\right)\left(\frac{2}{3} - \epsilon_2^2 + \frac{\epsilon_3}{3}\right). \end{cases} \tag{10.50}$$

We have to interpret ϵ_2 as fixed by the target problem and ϵ_3 as well as s_2 by the choice of numerical dissipation of the bulk Finite Difference scheme, *i.e.* the right hand sides in (10.50). Therefore, the unknowns (or the leverages) are s_3 and w_3, forming a non-linear system. Eliminating w_3 from the second equation in (10.50) using the first one yields—following some algebra—the equation for s_3:

$$(1-s_2)\left(\frac{2}{3} - \epsilon_2^2 + \frac{\epsilon_3}{3}\right)s_3 = (2-s_2)(1-s_2)\left(\frac{2}{3} - \epsilon_2^2 + \frac{\epsilon_3}{3}\right).$$

We have different cases to discuss which are summarized in Table 10.2.

- $s_2 = 1$. Then the equation is trivially satisfied for any choice of s_3. Enforcing the choice of $s_2 = 1$ in the first equation of (10.50) yields $(1-s_3)(w_3 - \epsilon_3) = 0$. This equation is trivially satisfied for $s_3 = 1$. If $s_3 \neq 1$, then we must initialize at equilibrium, that is, consider $w_3 = \epsilon_3$.
- $s_2 \neq 1$. Then the equation for s_3 reads $(2/3 - \epsilon_2^2 + \epsilon_3/3)s_3 = (2-s_2)(2/3 - \epsilon_2^2 + \epsilon_3/3)$. We distinguish two cases
  - $\epsilon_3 > -2+3\epsilon_2^2$. In this case, we have to enforce

$$s_3 = 2 - s_2. \tag{10.51}$$

    This is very interesting because it corresponds to the choice of "magic parameter" [d'Humières and Ginzburg, 2009, Kuzmin et al., 2011] equal to $1/4$, *cf.* Example 7.6.2. Using this choice of s_3 into the first equation from (10.50), we obtain that w_3 has to be taken as

$$w_3 = \frac{1}{s_2}(2(-2+3\epsilon_2^2) + (s_2-2)\epsilon_3). \tag{10.52}$$

    Remark that in this case, the only way of making the bulk scheme to be of second-order is to take $s_2 = 2$. This results in $s_3 = 0$, which means that one more moment is conserved by the scheme. Still, the equilibria do not depend on it. Moreover, the initialisation has to be $w_3 = -2+3\epsilon_2^2 \neq \epsilon_3$.

    In Section 7.6.3 we have found that the choice (10.52) could yield a bulk Finite Difference scheme with three stages instead of four. As far as the stability—*cf.* Theorem 7.7.3—under this condition is concerned, the analytical conditions in this case are

$$s_2 \in]0,2], \qquad \text{and} \qquad \begin{cases} |\epsilon_2| \leq 1, & -2+3\epsilon_2^2 \leq \epsilon_3 \leq 1, \quad \text{if} \quad s_2 \in]0,2[, \\ |\epsilon_2| < 1, & \quad \text{if} \quad s_2 = 2, \end{cases}$$

    see Appendix A.3, where in $-2+3\epsilon_2^2 \leq \epsilon_3 \leq 1$, the left constraint enforces non-negative dissipation (stability for small wave-numbers) whereas the right one concerns large wave-numbers.

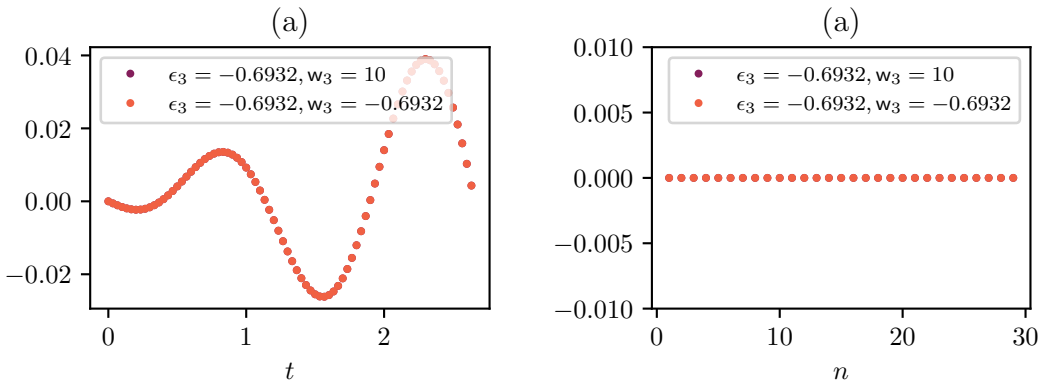


Figure 10.7: Left: test for smoothness in time close to $t = 0$ for the case (a) in Table 10.2: difference between exact and numerical solution at the eighth lattice point. As expected, regardless of the choice on w_3, the profile is smooth. Right: diffusion coefficient (factor in front of $-\lambda\Delta x\partial_{xx}$) in the modified equations for different n.

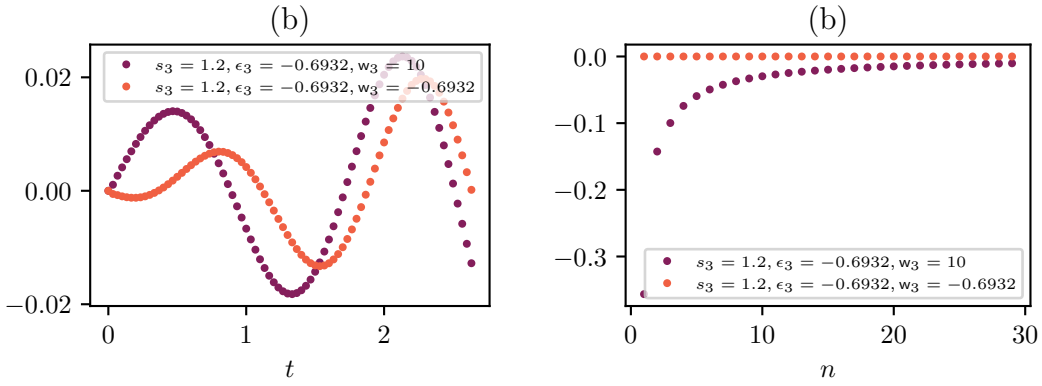


Figure 10.8: Left: test for smoothness in time close to $t = 0$ for the case (b) in Table 10.2 ($w_3 = -0.6932$) or violating this condition ($w_3 = 10$): difference between exact and numerical solution at the eighth lattice point. We observe radical differences in the profiles but the smoothness is not affected. Right: diffusion coefficient (factor in front of $-\lambda\Delta x\partial_{xx}$) in the modified equations for different n.

– $\epsilon_3 = -2 + 3\epsilon_2^2$. The equation is trivially true. Considering the first equation in (10.50) once more, we obtain $(1 - s_3)(w_3 + 2 - 3\epsilon_2^2) = 0$. If $s_3 = 1$, this equation is satisfied regardless of the choice of w_3. If $s_3 \neq 1$, then the initialisation should be $w_3 = -2 + 3\epsilon_2^2 = \epsilon_3$.

10.3.2.3 Study of the time smoothness of the numerical solution

We repeat the numerical experiment by [Van Leemput et al., 2009] introduced in Section 10.3.1.2. Only L^2 stable configurations are considered. As long as the dissipation of the bulk Finite Difference scheme is large, time oscillations are damped and thus cannot be observed even if the diffusivities of the bulk Finite Difference scheme and the initialisation schemes are not the same. We therefore look for situations where the numerical diffusion is small or zero.

- $s_2 = 1$, $\epsilon_3 = -2 + 3\epsilon_2^2$, no dissipation, and $s_3 = 1$. This is the framework of (a) (*cf.* Table 10.2), where we can consider arbitrary w_3. This case is trivial because $Q = 0$. We see in Figure 10.7 that the profile remains smooth no matter the choice of w_3, as predicted by the theory.
- $s_2 = 1$, $\epsilon_3 = -2 + 3\epsilon_2^2$, no dissipation, and $s_3 = 1.2$, close to one for stability reasons. Thus we are in the setting of (b). In Figure 10.8, we see that the choice of w_3 changes the outcome, even if the time smoothness seems

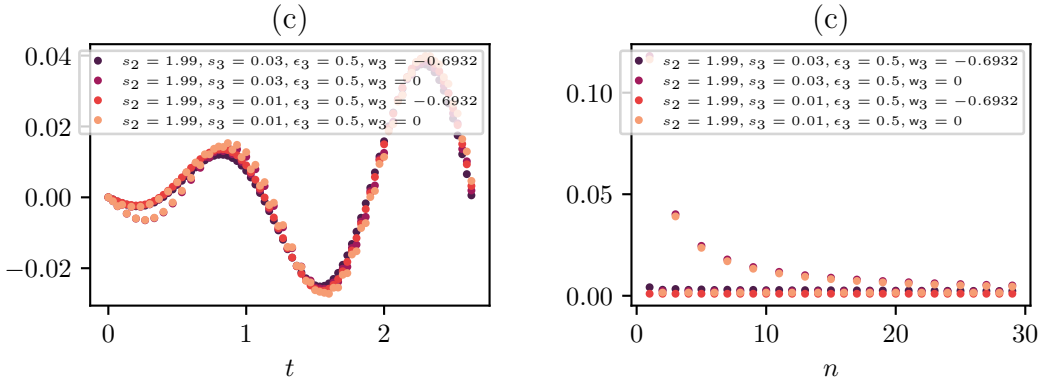


Figure 10.9: Left: test for smoothness in time close to $t = 0$ for the case (c) in Table 10.2: difference between exact and numerical solution at the eighth lattice point. The cases where $s_3 = 0.01$ violate the magic relation (10.51) $s_2 + s_3 = 2$ with minor influences on the spurious oscillation, whereas $w_3 = 0$ violates (10.52), with more tendency towards an initial boundary layer. Right: diffusion coefficient (factor in front of $-\lambda\Delta x\partial_{xx}$) in the modified equations for different n.

to be preserved in both cases. To explain this, on the one hand, we have to take into account that since we are compelled to take s_3 close to one, we are not far from the previous case. On the other hand, even when the dissipation is not matched, it does not oscillate between time steps, unlike many initialisations for the D_1Q_2 scheme in Section 10.3.1. This is confirmed by the right image in Figure 10.8: the diffusivity behaves smoothly in n and tends monotonically and quite rapidly to the bulk vanishing one.

- $s_2 = 1.99$, almost zero dissipation. We test (c), since (d) and (e) cannot be considered for stability reasons. In Figure 10.9, we observe that violating the magic relation (10.51) still enforcing (10.52) does not produce large spurious oscillations, likely because this has limited effects on the diffusion coefficient. Quite the opposite, violating (10.52) both with and without (10.51) produces an initial oscillating boundary layer. This is corroborated by the right image in Figure 10.9, where the reason for the observed oscillations is the highly non-smooth behaviour of the diffusion coefficient in n, as a result of having taken $s_2 \simeq 2$.

10.3.3 Conclusions

In Section 10.3, we have observed in practice that the conditions to obtain consistent starting schemes found in Section 10.2 preserve second-order convergence when the bulk scheme is second-order consistent. Using an additional order for the modified equations introduced in Section 10.2, we obtain an extremely precise description of the behaviour of the D_1Q_2 close to the initial time, according to the initialisation at hand. The same has been done for a D_1Q_3 scheme. Finally, discussing the conditions to have the same dissipation between initialisation and bulk schemes for the D_1Q_3 has made the magic relations (10.51) known in the literature [d'Humières and Ginzburg, 2009, Kuzmin et al., 2011] turn up once more, *cf.* Section 7.6.3. The investigation of these relations is central in the following Section 10.4.

10.4 A more precise evaluation of the number of initialisation schemes

In Section 10.2, we have observed that describing the behaviour of general lattice Boltzmann schemes close to the initial time above $O(\Delta x)$ order—using the modified equations—seems out of reach. The question which we try to answer here—inspired by the findings on the D_1Q_3 in Section 10.3.2—concerns the existence of vast classes of lattice Boltzmann schemes for which a detailed description of the behaviour of the initialisation schemes is indeed possible. The idea is to investigate the possibility of having, from a purely algebraic standpoint, a very small number of initialisation schemes to be considered. For example, this would allow to avoid dealing—when

trying to have the same dissipation coefficient between initialisation and bulk—with large non-linear systems such as (10.50), where the number and the complexity of equations grow with Q. The conditions to control the initialisation until a certain order in Δx could be simpler thanks to the fact that we have a small number of initialisation steps. In this way, if something similar to Proposition 10.2.3 was valid, we could conclude that this control is enough to master the dynamics of the scheme at the considered orders eventually in time.

10.4.1 Lattice Boltzmann schemes as dynamical systems and observability

A preliminary step in this direction is to consider any lattice Boltzmann scheme Algorithm 6 as a linear time-invariant discrete-time system

$$\begin{aligned} \mathsf{z}\mathbf{m}(t,\boldsymbol{x}) &= \mathbf{E}\mathbf{m}(t,\boldsymbol{x}), \qquad && (t,\boldsymbol{x}) \in \Delta t\mathbb{N} \times \Delta x\mathbb{Z}^d, \\ \mathbf{m}(0,\boldsymbol{x}) & && \text{given for } \boldsymbol{x} \in \Delta x\mathbb{Z}^d, \end{aligned}$$

where the output is $\mathbf{y} = \boldsymbol{C}\mathbf{m}$ with matrix $\boldsymbol{C}$ of appropriate dimension. Since, from the very beginning of the paper, we are solely interested in the conserved moment m_1, we select $\boldsymbol{C} = \boldsymbol{e}_1^{\mathsf{t}} \in \mathbb{R}^q$. As we already pointed out, see (10.8)

$$\mathsf{y}(n\Delta t,\boldsymbol{x}) = \mathsf{m}_1(n\Delta t,\boldsymbol{x}) = (\mathbf{E}^n\mathbf{m})_1(0,\boldsymbol{x}) = \boldsymbol{C}\mathbf{E}^n\mathbf{m}(0,\boldsymbol{x}), \qquad n \in \mathbb{N}, \quad \boldsymbol{x} \in \Delta x\mathbb{Z}^d,$$

thus we introduce the observability matrix of the system

$$\boldsymbol{\Omega} := \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{C}\mathbf{E} \\ \vdots \\ \boldsymbol{C}\mathbf{E}^{q-1} \end{bmatrix} \in \mathcal{M}_q(\mathsf{D}).$$

If the system were set on a field (*e.g.* $\boldsymbol{\Omega} \in \mathcal{M}_q(\mathbb{R})$ or $\boldsymbol{\Omega} \in \mathcal{M}_q(\mathbb{C})$), it would be customary to call the system "observable" if and only if $\mathrm{rank}(\boldsymbol{\Omega}) = q$. This would mean that we could reconstruct the initial data $\mathbf{m}(0)$ from the observation of $\mathsf{y} = \mathsf{m}_1$ at times $n \in [\![0, q-1]\!]$. Quite the opposite, in our case, since the non-zero entries of $\boldsymbol{\Omega}$ are in general not invertible (for $d = 1$, the symmetric $\mathsf{S}(\mathsf{x}_1)$ and anti-symmetric part $\mathsf{A}(\mathsf{x}_1)$ of the basic shift are examples of this), we cannot proceed in the same way, because the observability matrix $\boldsymbol{\Omega}$ can never be a unit.

For systems over commutative rings, different definition of observability are available in the literature: we list a few of them in the following Definition.

Definition 10.4.1: Observability for systems on rings

The system is said to be

- "observable" according to [Brewer et al., 1986, Theorem 2.6], if the application represented by the left action of $\boldsymbol{\Omega}$ is injective.
- "observable" according to [Fliess and Mounier, 1998], if $\boldsymbol{\Omega}$ has left inverse.
- "hyper-observable" according to [Fliess and Mounier, 1998], if the unobservable sub-space $\mathcal{N} := \ker(\boldsymbol{\Omega})$— where operators act on lattice functions[a]—is trivial: $\mathcal{N} = \{\mathbf{0}\}$.

[a] Observe that the kernel is the left null space: indeed the left action of elements in D can operate both on lattice function and operators in D, whereas the right action is reserved for operators in D.

Furthermore, [Brewer et al., 1986, Theorem 2.6] gives the following criterion to check observability.

Theorem 10.4.1: Observability criterion

The system is "observable" according to [Brewer et al., 1986] if and only if the ideal of D generated by $\det(\boldsymbol{\Omega})$ is such that its annihilator is zero.

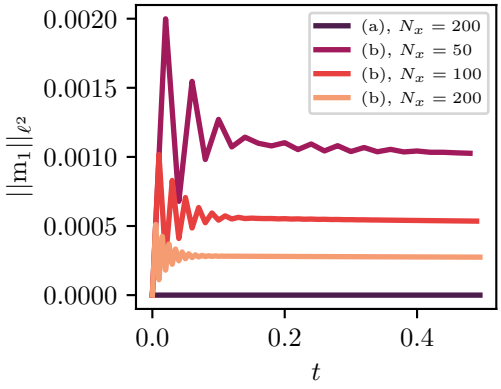


Figure 10.10: L^2 norm of the conserved moment as function of the time for the $\mathrm{D_1Q_2}$ scheme choosing $\lambda = 1$, $\epsilon_2 = 1/2$ and $s_2 = 1.8$. The test is performed for different initial data (both observable and unobservable) with different Δx.

We also define the "observability index" $o \leq Q+1$ mimicking the definition for systems over fields as

$$o := \max_{r\in\mathbb{N}} \operatorname{rank}(\boldsymbol{\Omega}_r), \quad \text{where} \quad \boldsymbol{\Omega}_r := \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{C}\mathbf{E} \\ \vdots \\ \boldsymbol{C}\mathbf{E}^{r-1} \end{bmatrix} \in \mathcal{M}_{r\times q}(\mathsf{D}),$$

and rank$(\cdot)$ stands for the row rank of a matrix over a ring according to [Blyth, 2018, Definition 10.6].

Example 10.4.1. *Considering Example 10.1.1 treated in Section 10.3.1, we have that*

$$\boldsymbol{\Omega} = \begin{bmatrix} 1 & 0 \\ \mathsf{S}(\mathsf{x}_1) + s_2\epsilon_2\mathsf{A}(\mathsf{x}_1) & (1-s_2)\mathsf{A}(\mathsf{x}_1) \end{bmatrix},$$

hence $o = Q+1 = 2$ if $s_2 \neq 1$ and $o = Q+1 = 1$ if $s_2 = 1$. When $s_2 = 1$, we have the unobservable subspace $\mathcal{N} = \{(0, \mathsf{m}_2)^\mathsf{t} \ : \ \text{for arbitrary } \mathsf{m}_2 = \mathsf{m}_2(x) \text{ lattice function}\}$, which adheres to the intuition that we cannot know the non-conserved moment m_2 by looking at the conserved moment m_1 if the relaxation is made on the equilibrium, regardless of the structure of m_2. When $s_2 \neq 1$, we have $\mathcal{N} = \{(0, \mathsf{m}_2)^\mathsf{t} \ : \ \text{for any } \mathsf{m}_2 = \mathsf{m}_2(x) \text{ lattice function such that } \mathsf{A}(\mathsf{x}_1)\mathsf{m}_2 = 0\}$. We see that the unobservable sub-space is non-trivial even when $o = q = 2$, contrarily to the case of systems with matrix $\mathbf{E}$ and $\boldsymbol{\Omega}$ with entries in a field. The unobservable states are those in which the first component is zero and the discrete derivative $\mathsf{A}(\mathsf{x}_1)$ of the second component is zero everywhere, for example because the second component is constant or takes one given value on all even point and another one on all odd point.

To numerically check the structure of $\mathcal{N}$ for this scheme, we consider two sets of initial data

$$(a) \qquad \mathsf{m}_1(0,\cdot) = 0, \qquad \mathsf{m}_2(0, j\Delta x) = \frac{1+3(-1)^j}{8},$$

$$(b) \qquad \mathsf{m}_1(0,\cdot) = 0, \qquad \mathsf{m}_2(0, j\Delta x) = \frac{1}{10}\exp\Big(-\frac{1}{1-(4(j\Delta x - 1/2))^2}\Big).$$

The first datum (a) lies in $\mathcal{N}$ whereas the second one (b) does not. Observe that both data do not adhere to the guidelines to choose initial data according to the analysis in Section 10.2: they are uniquely selected for the current test. We shall take $j \in [\![0, N_x - 1]\!]$ in the simulations and $\Delta x = 1/N_x$. Periodic boundary conditions are enforced. The results of the simulation given in Figure 10.10 confirm the theory. The unobservable initial datum (a) yields zero conserved (observed) moment for any time step, whereas the observable one (b) does not, even if the conserved moment is initialized as zero everywhere. For the observable datum (b), we see that the solution converges linearly in Δx to the exact solution of the Cauchy problem, meaning the identically zero solution.

We finally comment on the notions from Definition 10.4.1.

- $\det(\boldsymbol{\Omega}) = (1-s_2)\mathsf{A}(\mathsf{x}_1)$, *thus the ideal to consider (*cf. *Theorem 10.4.1) is* $\{\mathsf{d}(1-s_2)\mathsf{A}(\mathsf{x}_1) \ : \ \mathsf{d} \in \mathsf{D}\}$. *On the one hand, if* $s_2 = 1$, *then any operator in* D *multiplied at the left of any element of the ideal is an annihilator, thus the system is not observable according to [Brewer et al., 1986]. On the other hand, if* $s_2 \neq 1$, *then the only element annihilating any element of the ideal is zero, thus the system is observable according to [Brewer et al., 1986].*
- *For any* s_2, *we see that* $\boldsymbol{\Omega}$ *does not admit left inverse, therefore it is not observable according to [Fliess and Mounier, 1998].*
- *For any* s_2, *the system is not hyper-observable according to [Fliess and Mounier, 1998] due to the non-trivial* $\mathcal{N}$.

For these reasons, we infer that the observability according to [Brewer et al., 1986] is the one more closely adhering—between those issued from Definition 10.4.1—to our definition of observability index o.

10.4.2 Reduced number of initialisation schemes for non-observable systems

Following the discussion in Section 7.6.3 and in particular (7.35), we can introduce $\mathbf{p}_o \in \mathsf{D}^o$ such that

$$\mathbf{p}_o \boldsymbol{\Omega}_o = -\boldsymbol{C}\mathbf{E}^o. \tag{10.53}$$

The solution of this problem exists thanks to the definition of the observability index o. We then introduce the monic polynomial, in the spirit of Definition 7.6.3 and Lemma 7.6.2 (keep in mind that the indices in vectors like $\mathbf{p}_o$ start from one)

$$\Psi_o(\mathsf{z}) := \mathsf{z}^o + \sum_{k=1}^{o} \mathsf{p}_{o,k}\mathsf{z}^{k-1}, \tag{10.54}$$

which by construction (10.53) annihilates the first row of $\mathbf{E}$, since $\boldsymbol{C} = \boldsymbol{e}_1^{\mathsf{t}}$. Moreover, we have shown in Lemma 7.6.3 (just apply the matrix-determinant Lemma 8.2.1 to conclude) that $\Psi_o(\mathsf{z})$ divides $\det(\mathsf{z}\boldsymbol{I}-\mathbf{E})$, whence if $o = Q+1$, we naturally have $\Psi_o(\mathsf{z}) = \mathsf{z}^{Q+1-q}\det(\mathsf{z}\boldsymbol{I}-\mathbf{E})$. We therefore obtain the following corresponding bulk Finite Difference scheme based on Ψ_o given by Algorithm 8, coinciding with Algorithm 7 when $o = Q+1$.

Algorithm 8 Corresponding Finite Difference scheme based on Ψ_o.

- Given $\mathbf{m}(0,\boldsymbol{x})$ for every $\boldsymbol{x} \in \Delta x\mathbb{Z}^d$.
- **Initialisation schemes**. For $n \in [\![1, o-1]\!]$

$$\mathsf{m}_1(n\Delta t, \boldsymbol{x}) = \boldsymbol{C}\mathbf{E}^n\mathbf{m}(0,\boldsymbol{x}), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d. \tag{10.55}$$

- **Corresponding bulk Finite Difference scheme**. For $n \in [\![o-1, +\infty[\![$

$$\mathsf{m}_1((n+1)\Delta t, \boldsymbol{x}) = -\sum_{k=q-o}^{q-1} \mathsf{p}_{o,o+k+1-q}\mathsf{m}_1((n+k+1-q)\Delta t, \boldsymbol{x}), \qquad \boldsymbol{x} \in \Delta x\mathbb{Z}^d. \tag{10.56}$$

The lack of observability is indeed the reason why, as previously seen in Section 10.3.2, one can find a bulk Finite Difference scheme with less time steps than what is prescribed by the characteristic polynomial of $\mathbf{E}$. From a different perspective, this is the so-called "pole-zero cancellation" in the transfer function—see for example [Åström and Murray, 2008, Chapter 8.3] or in [Hendricks et al., 2008, Chapter 3.9]—from control theory. In our framework, the transfer function is

$$\mathsf{H}(\mathsf{z}) = \boldsymbol{C}\frac{\overbrace{\operatorname{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{B}\boldsymbol{\epsilon}}^{\text{control by equil.}}}{\underbrace{\det(\mathsf{z}\boldsymbol{I}-\mathbf{A})}_{\text{state}}} = \boldsymbol{C}(\mathsf{z}\boldsymbol{I}-\mathbf{A})^{-1}\mathbf{B}\boldsymbol{\epsilon},$$

where we recall that $\mathbf{A} = \mathbf{T}(\boldsymbol{I}-\boldsymbol{S})$ and $\mathbf{B} = \mathbf{T}\boldsymbol{S}$, *cf.* Section 7.5.

Example 10.4.2. *We come back to the scheme of Section 10.3.2 where we have selected the choice of magic parameter equal to* $1/4$*, that is* $s_2 + s_3 = 2$*. We also assume that* $s_2 \neq 1$ *to keep things non-trivial. In this case, it can be seen that* $o = 2 < 3$*, whereas* $Q + 1 = 3$*. Moreover, we obtain*

$$\begin{aligned}
&\det(\mathsf{z}\boldsymbol{I} - \mathbf{E}) = (\mathsf{z} + (1 - s_2))\Psi_2(\mathsf{z}),\\
&\qquad \textit{with} \quad \Psi_2(\mathsf{z}) = \mathsf{z}^2 + (-s_2\epsilon_2\mathsf{A}(\mathsf{x}_1) + \tfrac{1}{3}(s_2 - 2)(2\mathsf{S}(\mathsf{x}_1) + 1) + \tfrac{1}{3}\epsilon_3(s_2 - 2)(\mathsf{S}(\mathsf{x}_1) - 1))\mathsf{z} + (1 - s_2),\\
&\qquad \textit{or equivalently} \quad \mathsf{H}(\mathsf{z}) = \frac{(\mathsf{z} + (1 - s_2))(s_2\epsilon_2\mathsf{A}(\mathsf{x}_1) + \frac{1}{3}\epsilon_3(2 - s_2)(\mathsf{S}(\mathsf{x}_1) - 1))\mathsf{z}}{(\mathsf{z} + (1 - s_2))\underbrace{(\mathsf{z}^2 + \frac{1}{3}(s_2 - 2)(2\mathsf{S}(\mathsf{x}_1) + 1)\mathsf{z} + (1 - s_2))}_{\text{called } \Psi_{\mathbf{A}}(\mathsf{z}) \text{ in Section 7.6.3}}}.
\end{aligned}$$

The Finite Difference scheme coming from $\Psi_2(\mathsf{z})$ *becomes a leap-frog scheme for* $s_2 = 2$*. Otherwise, it is a centered discretisation with a certain amount of numerical dissipation. A first question which might arise concerns the modified equation for the bulk Finite Difference scheme obtained using* $\det(\mathsf{z}\boldsymbol{I} - \mathbf{E})$*, see Algorithm 7,* versus *those obtained by* $\Psi_2(\mathsf{z})$*, see Algorithm 8. The answer is that they are same at any order because the eigenvalue* $(s_2 - 1)$ *does not contribute to the consistency (being constant through wave-numbers and thus being a mere numerical eigenvalue) and it can be easily checked that* $\Psi_2(\mathsf{z})$ *yields the same modified equation, since it contains the consistency eigenvalue [Strikwerda, 2004]. As far as stability is concerned, the stability constraints for the two bulk Finite Difference schemes are the same because* $|s_2 - 1| \leq 1$ *for* $s_2 \in]0, 2[$*. The stability conditions are analytically computed in Appendix A.3. The case* $s_2 = 2$ *might produce instabilities because of the presence of multiple roots of* $\det(\mathsf{z}\boldsymbol{I} - \hat{\mathbf{E}})$ *on the unit circle. However, in this case, there is an additional conserved moment* m_3 *and we know that the* von Neumann *condition for systems is that no root is outside the unit circle, but this is only necessary for stability, see Theorem 9.3.2. Therefore, the presence of multiple eigenvalues on the unit circle (and in particular those concerning consistency which are now more than one) cannot allow to deduce that the scheme is unstable.*

Concerning the notion of observability by [Brewer et al., 1986], we have that $\det(\boldsymbol{\Omega}) = 0$*, hence the system is not observable, according to Theorem 10.4.1. If we want to characterize the unobservable sub-space, we have that, since* $s_2 \neq 1$*, it is given by*

$$\mathcal{N} = \{(0, \mathsf{m}_2, \mathsf{m}_3)^{\mathsf{t}} \ : \ \textit{for any } \mathsf{m}_2 = \mathsf{m}_2(x),\ \mathsf{m}_3 = \mathsf{m}_3(x) \textit{ lattice fnct. such that } \mathsf{A}(\mathsf{x}_1)\mathsf{m}_2 = \tfrac{1}{3}(\mathsf{S}(\mathsf{x}_1) - 1)\mathsf{m}_3\}.$$

Recall that $\mathsf{A}(\mathsf{x}_1) = (\mathsf{x}_1 - \mathsf{x}_1^{-1})/2$ *and* $\mathsf{S}(\mathsf{x}_1) - 1 = (\mathsf{x}_1 - 2 + \mathsf{x}_1^{-1})/2$*, which means that the initial states belonging to* $\mathcal{N}$ *are those with zero first moment everywhere and such that the centered approximation of the first derivative of the second moment is proportional—with ratio* $1/3$*—to the centered approximation of the second derivative of the third moment, at any point of the lattice.*

The numerical verification of the expression found for $\mathcal{N}$ *can be done as follows. We select*

$$\mathsf{m}_1(0, \cdot) = 0, \qquad \mathsf{m}_2(0, j\Delta x) = j, \qquad \mathsf{m}_3(0, j\Delta x) = -3j^2,$$

which thus belongs to $\mathcal{N}$*. We discretize with* $j \in [\![0, 100[\![$ *using periodic boundary conditions. These boundary conditions are incompatible with the data, but we shall observe the outcome way inside the computational domain. The result of the simulation is proposed in Figure 10.11. For the choice where* $s_3 = 2 - s_2$*, thus for which the initial datum belongs to* $\mathcal{N}$*, we see that away from the boundary, the conserved moment remains zero (up to machine precision). When* $s_3 \neq 2 - s_2$*, thus the initial datum is observable, we remark that even inside the domain, the conserved moment is non-zero (around* 0.383*).*

As already remarked in [Saad, 1989] and Section 7.6.3, the cases were $Q + 1 \neq o$ are extremely peculiar. Indeed, the situation described in Example 10.4.2 and in the forthcoming Section 10.4.3 are the only examples we were able to find. Loosely speaking, both o and Q measure the speed of saturation of the image of the scheme $\mathbf{E}$ concerning the conserved moment. Once the generated sub-spaces saturate, the evolution of the conserved moment at the new time-step can be recast as function of itself at the previous steps. The fact that the Cayley-Hamilton theorem holds (concerning Q) and that the polynomial Ψ_o (concerning o) annihilates the first row of $\mathbf{E}$ introduce—as previously shown—a set of linear constraints on m_1, solution of the lattice Boltzmann scheme.

It should be emphasized that Proposition 10.2.3 is still valid turning Q into $o - 1$, (10.8) into (10.55) and (10.9)

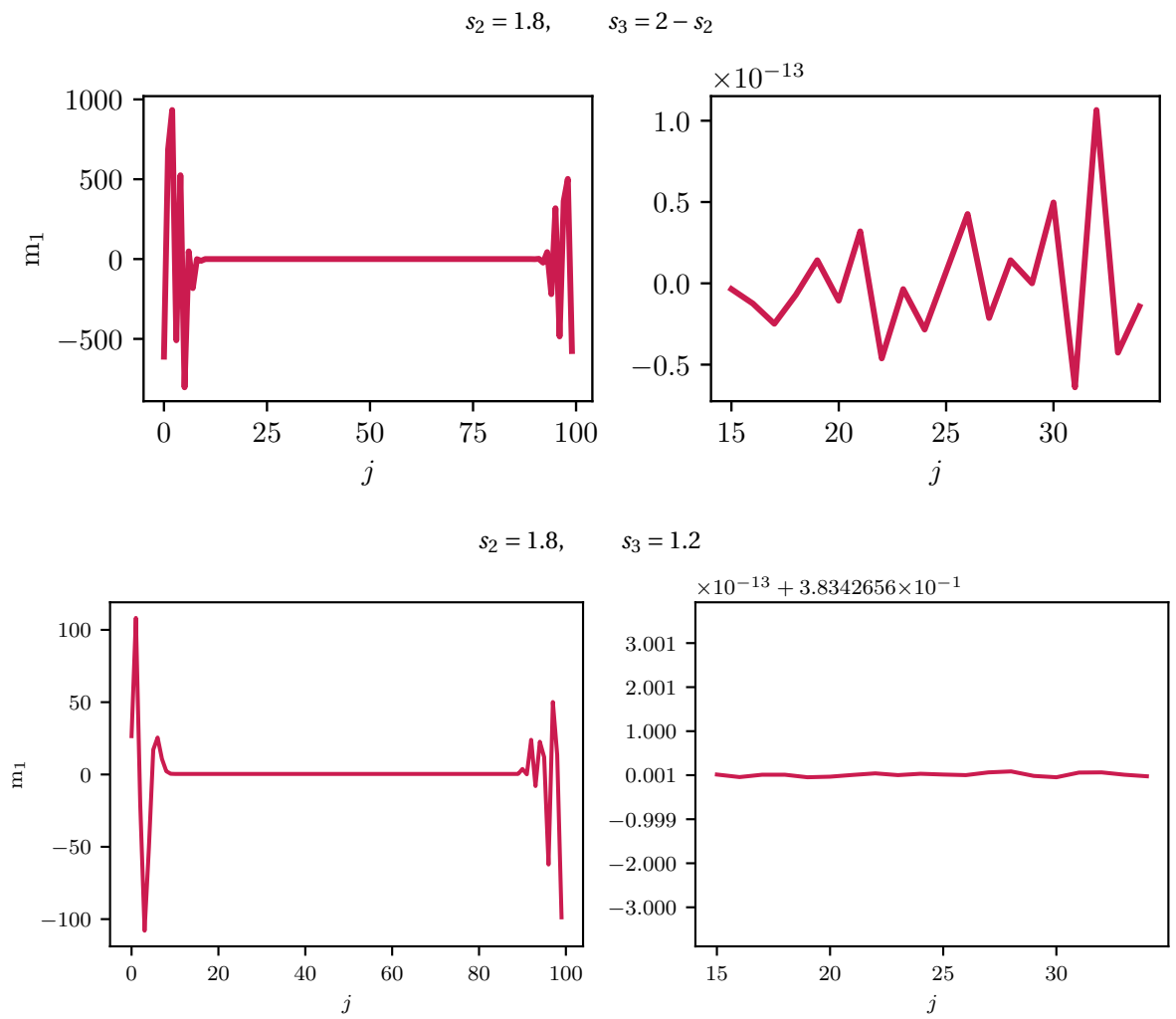


Figure 10.11: Conserved moment after 10 iterations for the D_1Q_3 scheme choosing $\lambda = 1, \epsilon_2 = 0.5, \epsilon_3 = 0.1, N_x = 100$ and different relaxation parameter.

into (10.56). This is fundamental, because Proposition 10.2.3 ensures to control the whole dynamics of the scheme by mastering it in the initialisation layer. The aim of studying observability is to characterise in which case the initialisation layer (10.55), thus what we need to control, is simple but still determines the dynamics eventually in time. This property comes from the fact that the root of $\det(z\boldsymbol{I} - \hat{\mathbf{E}}(\boldsymbol{\xi}\Delta x))$ setting the consistency of the scheme—*i.e.* being one in the low-frequency limit—is also a root of $\hat{\Psi}_o(z)$. This is a consequence of the fact that $\hat{\Psi}_o(z)$ annihilates the first row of $\hat{\mathbf{E}}(\boldsymbol{\xi}\Delta x)$.

Proposition 10.4.1

Let $\hat{g}_1 \equiv \hat{g}_1(\boldsymbol{\xi}\Delta x)$ be the unique root of $\det(z\boldsymbol{I} - \hat{\mathbf{E}}(\boldsymbol{\xi}\Delta x))$ such that

$$\hat{g}_1(\boldsymbol{\xi}\Delta x) = 1 + O(|\boldsymbol{\xi}\Delta x|) \tag{10.57}$$

in the limit $|\boldsymbol{\xi}\Delta x| \ll 1$, which is the one determining the consistency and the modified equation of the numerical scheme. Then, $\hat{g}_1$ is also a root of $\hat{\Psi}_o(z)$.

Proof. Let us show this, using the Fourier representation and considering $d = 1$ for the sake of keeping notations

simple. Recall that

$$\hat{\Psi}_o(\hat{\mathbf{E}}(\xi\Delta x)) = \hat{\mathbf{E}}(\xi\Delta x)^o + \sum_{k=1}^{o} \hat{\mathrm{p}}_{o,k}(\xi\Delta x)\hat{\mathbf{E}}(\xi\Delta x)^{k-1} = \begin{bmatrix} 0 & \cdots & 0 \\ \star & \cdots & \star \\ \vdots & & \vdots \\ \star & \cdots & \star \end{bmatrix}, \tag{10.58}$$

where the starred $\star$ entries are not necessarily zero. Notice that, whatever the scaling between space and time, we have that for every $r \in \mathbb{N}$

$$\hat{\mathbf{E}}(\xi\Delta x)^r = \begin{bmatrix} 1 & \cdots & 0 \\ \star & \cdots & \star \\ \vdots & & \vdots \\ \star & \cdots & \star \end{bmatrix} + O(|\xi\Delta x|) \tag{10.59}$$

in the limit $|\xi\Delta x| \ll 1$. In particular, for the acoustic scaling, we have $\hat{\mathbf{E}}(\xi\Delta x)^r = \boldsymbol{K}^r + O(|\xi\Delta x|)$, where $\boldsymbol{K}^r$ has the property stated by (10.59) and $\boldsymbol{K}$ is the collision matrix. Again taking $|\xi\Delta x| \ll 1$ and considering that $\hat{\mathrm{p}}_{o,k}(\xi\Delta x) = \hat{p}^{(0)}_{o,k} + O(|\xi\Delta x|)$, selecting the very first entry in (10.58) yields, using (10.59)

$$1 + \sum_{k=1}^{o} \hat{p}^{(0)}_{o,k} = O(|\xi\Delta x|). \tag{10.60}$$

Since $\mathbb{C}$ is an algebraically closed field, we can write $\hat{\Psi}_o(\mathrm{z}) = \prod_{r=1}^{r=o}(\mathrm{z} - \hat{\mathrm{r}}_r(\xi\Delta x))$, where $\hat{\mathrm{r}}_r$ for $r \in [\![1, o]\!]$ are the roots of $\hat{\Psi}_o(\mathrm{z})$. These are also part of the roots of $\det(\mathrm{z}\boldsymbol{I} - \hat{\mathbf{E}}(\xi\Delta x))$ since $\hat{\Psi}_o(\mathrm{z})$ divides $\det(\mathrm{z}\boldsymbol{I} - \hat{\mathbf{E}}(\xi\Delta x))$. The question is whether the roots of $\hat{\Psi}_o(\mathrm{z})$ include the one of $\det(\mathrm{z}\boldsymbol{I} - \hat{\mathbf{E}}(\xi\Delta x))$, indicated by $\hat{\mathrm{g}}_1(\xi\Delta x)$, being the only one such that $\hat{\mathrm{g}}_1(\xi\Delta x) = 1 + O(|\xi\Delta x|)$ in the limit $|\xi\Delta x| \ll 1$, see (10.26), and which totally dictates consistency (and the modified equations). Considering $\mathrm{z} = 1$ in (10.54) gives

$$\hat{\Psi}_o(1) = 1 + \sum_{k=1}^{o} \hat{\mathrm{p}}_{o,k}(\xi\Delta x).$$

Taking the limit $|\xi\Delta x| \ll 1$, we are left with

$$\prod_{r=1}^{o}(1 - \hat{r}^{(0)}_r) + O(|\xi\Delta x|) = 1 + \sum_{k=1}^{o} \hat{p}^{(0)}_{o,k} + O(|\xi\Delta x|) = O(|\xi\Delta x|),$$

thanks to (10.60), where $\hat{\mathrm{r}}_r = \hat{r}^{(0)}_r + O(|\xi\Delta x|)$. This gives $\prod_{r=1}^{r=o}(1 - \hat{r}^{(0)}_r) = 0$, hence at least one $\hat{r}^{(0)}_r = 1$. Since the roots $\hat{\mathrm{r}}_r$ for $r \in [\![1, o]\!]$ are a subset of those of $\det(\mathrm{z}\boldsymbol{I} - \hat{\mathbf{E}}(\xi\Delta x))$, where only one has the desired property (10.57), then the latter is also a root of $\hat{\Psi}_o(\mathrm{z})$, let us say $\hat{\mathrm{r}}_1 \equiv \hat{\mathrm{g}}_1$. □

10.4.3 An important case: link $\mathrm{D}_d\mathrm{Q}_{1+2W}$ two-relaxation-times schemes with magic parameters equal to 1/4

We are now ready to consider a quite wide class of schemes [d'Humières and Ginzburg, 2009] for which very little initialisation schemes are to consider, namely o is very small. The "observable" features of these schemes are to some extent independent from d and the choice of the $q = 1 + 2W$ discrete velocities. This boils down to a quite general application of the ideas of Section 10.4.2.

10.4.3.1 Description of the schemes

The schemes are exactly those of Example 7.6.2, with one zero velocity and the remaining $2W$ ones which are pairwise opposite, with $N = 1$. Again, the relaxation paramters are $s_{2r} = s$ and $s_{2r+1} = 2 - s$ for $r \in [\![1, W]\!]$ with

$s \in]0,2]$. The only difference here is that we consider the dimensionless form of the moment matrix

$$\boldsymbol{M} = \left[\begin{array}{c|ccc|c|c} 1 & 1 & 1 & \cdots & 1 & 1 \\ \hline 0 & 1 & -1 & & & \\ 0 & 1 & 1 & & & \\ \hline \vdots & & & \ddots & & \\ \hline 0 & & & & 1 & -1 \\ 0 & & & & 1 & 1 \end{array}\right] \in \mathcal{M}_{1+2W}(\mathbb{R}), \tag{10.61}$$

in order to analyze these schemes both under acoustic and diffusive scaling.

10.4.3.2 Observability and number of initialisation steps

The study of the observability of the previously described schemes is carried in the following result.

Proposition 10.4.2

The characteristic polynomial of the scheme matrix $\mathbf{E}$ for the schemes given in Example 7.6.2 with (10.61) is given by

$$\det(z\boldsymbol{I} - \mathbf{E}) = (z + (1-s))(z^2 - (1-s)^2)^{W-1}\Psi_2(z),$$

where

$$\Psi_2(z) = z^2 + (s-2)z + (1-s) - zs\sum_{r=1}^{W} A(t_{\boldsymbol{c}_{2r}})\epsilon_{2r} + z(s-2)\sum_{r=1}^{W}(S(t_{\boldsymbol{c}_{2r}}) - 1)\epsilon_{2r+1} \tag{10.62}$$

annihilates the first row of the matrix $\mathbf{E}$. Therefore $o = 2$ if $s \neq 1$ and $o = 1$ if $s = 1$. Equivalently, the transfer function of the system is given by

$$H(z) = \frac{(z + (1-s))(z^2 - (1-s)^2)^{W-1}\left(s\sum_{r=1}^{r=W} A(t_{\boldsymbol{c}_{2r}})\epsilon_{2r} + (2-s)\sum_{r=1}^{r=W}(S(t_{\boldsymbol{c}_{2r}}) - 1)\epsilon_{2r+1}\right)z}{(z + (1-s))(z^2 - (1-s)^2)^{W-1}\underbrace{(z^2 + (s-2)z + (1-s))}_{\text{called } \Psi_{\mathbf{A}}(z) \text{ in Section 7.6.3}}}.$$

By Proposition 10.4.2, $\Psi_2(z)$ yields a bulk Finite Difference scheme according to Algorithm 8. As for Example 10.4.2, the modified equations of the method obtained by $\Psi_2(z)$ and by $\det(z\boldsymbol{I} - \mathbf{E})$ are the same because the remaining roots do not concern consistency. The only consistency eigenvalue is one of the two roots of $\Psi_2(z)$, thus present in both schemes. The case $s = 2$ apparently questions the previous claim since by looking at the proof of Proposition 10.2.3, a scheme consistent with (10.1) has only one eigenvalue equal to one for small wave-numbers. This is not a contradiction, because in this case $s_{2r+1} = 0$ for $r \in [\![1, W]\!]$, thus the corresponding moments are conserved [Ginzburg et al., 2008b], whereas Theorem 10.2.1 has been demonstrated under the assumption that $s_i \neq 0$ for $i \in [\![2, q]\!]$ and the whole chapter relies on the assumption that we deal only with one conserved moment. The moments m_{2r+1} for $r \in [\![1, W]\!]$ are conserved not because their equilibrium satisfy (1.3), but rather since their corresponding relaxation parameter is zero. A valid proof of Theorem 10.2.1 for several conserved moments has to follow the indications of Chapter 7 and Chapter 8 and would still lead to (10.10). Using Theorem 8.3.1 with $N = 1 + W$, since $s = 2$, we would get, for the first moment

$$\partial_t \phi(t, \boldsymbol{x}) + \lambda \sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \phi(t, \boldsymbol{x}) = O(\Delta x). \tag{10.63}$$

For the conserved moments m_{2r+1} for $r \in [\![1, W]\!]$, we obtain

$$\partial_t m_{2r+1}(t, \boldsymbol{x}) + \lambda \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \phi(t, \boldsymbol{x}) = O(\Delta x). \tag{10.64}$$

Observe that the equation (10.63) for the moment of interest is indeed independent of the other conserved moments, as desired (no possible coupling *via* the equilibria, for they depend only on the first conserved moment). Quite the opposite, the equations (10.64) for the "inadvertently" conserved moments couple them with the first

one. The first conserved moment is going to evolve alone, as usual, and the dynamics of the other conserved moments is going to be coupled with the one of m_1 according to (10.64). Still, we are not interested in the latter moments. Coming back to our case, where we operate as only one moment was conserved even when $s=2$, the multiplicative factor $(\mathsf{z}+(1-s))(\mathsf{z}^2-(1-s)^2)^{W-1} \asymp (\exp(\Delta x/\lambda\partial_t)-1)(\exp(2\Delta x/\lambda\partial_t)-1)^{W-1} = O(\Delta x^W)$ in front of the amplification polynomial Ψ_2 of a leap-frog scheme is a series of time differential operators starting with a term of kind $\Delta x^W \partial_t^W$. Thus, as observed in Example 7.5.6, if we compute the modified equation of the corresponding Finite Difference scheme obtained as we were dealing only with one conserved moment (*i.e.* (10.6)) whereas several conserved moments are present, we would obtain a sort of wave equation with time derivative of order $1+W$. This is unsurprising since this kind of equation feature $1+W$ "consistency" eigenvalues (one of which, the actual one, is inside $\Psi_2(\mathsf{z})$) which values equal one for small wave-numbers.

Coming back to generic s, the stability conditions of the corresponding Finite Difference obtained by using $\Psi_2(\mathsf{z})$ instead of $\det(\mathsf{z}\boldsymbol{I}-\mathbf{E})$ are the same because the remaining roots are constant in wave-number and do not exceed modulus one when $s\in]0,2[$. The case $s=2$ might produce instabilities because of the presence of multiple roots of $\det(\mathsf{z}\boldsymbol{I}-\hat{\mathbf{E}})$ on the unit circle. However, in this case, there are additional conserved moments and the *von Neumann* condition for systems is that no root is outside the unit circle, with no precision concerning multiple ones on the unit circle. Still this condition is only necessary for stability, see Theorem 9.3.2. Therefore, the presence of multiple eigenvalues on the unit circle (and in particular those concerning consistency which are now $1+W$) cannot allow to deduce that the scheme is unstable. This should be precisely tested in the case where $W\geq 2$, for example, taking a $\mathrm{D}_1\mathrm{Q}_5$ scheme.

Since $\det(\boldsymbol{\Omega})=0$, the system is not observable according to [Brewer et al., 1986]. However, it is not easy to generally characterize the unobservable sub-space $\mathcal{N}$, because this sub-space inflates with d and W due to the rank-nullity theorem. To explain this difficulty, consider that Ψ_2 from (10.62) is essentially scheme independent and concerns the "observable" part of the system relative to $\mathrm{span}(\boldsymbol{\Omega})$, whereas $\mathcal{N}=\ker(\boldsymbol{\Omega})$ must be highly scheme dependent because it pertains to the remaining "unobservable" part of the system, which is encoded in the quotient $\det(\mathsf{z}\boldsymbol{I}-\mathbf{E})/\Psi_2(\mathsf{z})$ between polynomials.

Let us now proceed to the proof of Proposition 10.4.2. The stream matrix is given by

$$\mathbf{T}=\left[\begin{array}{c|cc|c|cc}
1 & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2})-1 & \cdots & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}})-1 \\ \hline
0 & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\
0 & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\ \hline
\vdots & & & \ddots & & \\ \hline
0 & & & & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) \\
0 & & & & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2W}}) & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2W}})
\end{array}\right] \in \mathcal{M}_{1+2W}(\mathsf{D}).$$

Proof of Proposition 10.4.2. We have that $\det(\mathsf{z}\boldsymbol{I}-\mathbf{E}) = \det(\mathsf{z}\boldsymbol{I}-\mathbf{A}-\mathbf{B}\boldsymbol{\epsilon}\otimes\boldsymbol{e}_1) = \det(\mathsf{z}\boldsymbol{I}-\mathbf{A}) - \boldsymbol{e}_1^{\mathsf{t}}\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{B}\boldsymbol{\epsilon}$, using the matrix determinant Lemma 8.2.1. Also

$$\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})_{11} = \prod_{r=1}^{W}\det(\mathsf{z}\boldsymbol{I}-\tilde{\mathbf{T}}_r\,\mathrm{diag}(1-s,s-1)) = \prod_{r=1}^{W}(\mathsf{z}^2-(1-s)^2) = (\mathsf{z}^2-(1-s)^2)^W,$$

with

$$\tilde{\mathbf{T}}_r = \begin{bmatrix} \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}}) & \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}}) \\ \mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}}) & \mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}}) \end{bmatrix}.$$

We only treat $\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})_{12}$ and $\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})_{13}$, since the following entries read the same except for the indices of the involved shift operators.

$\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})_{12}$

$$= -\det \left[\begin{array}{cc|c|c|c} (s-1)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & (1-s)(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2})-1) & \star & \cdots & \star \\ (s-1)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_2}) & \mathsf{z}+(1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_2}) & & & \\ \hline & & \mathsf{z}\boldsymbol{I}-\tilde{\mathbf{T}}_2\mathrm{diag}(1-s,s-1) & \star & \star \\ \hline & & & \ddots & \star \\ \hline & & & & \mathsf{z}\boldsymbol{I}-\tilde{\mathbf{T}}_W\mathrm{diag}(1-s,s-1) \end{array}\right],$$

where the $\star$ blocks are not necessarily zero but do not need to be further characterized, since this is a determinant of a block upper triangular matrix, whence

$$\begin{aligned}\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})_{1,2r} &= -(\mathsf{z}^2-(1-s)^2)^{W-1}\det\begin{bmatrix}(s-1)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}}) & (1-s)(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})-1)\\ (s-1)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}}) & \mathsf{z}+(1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\end{bmatrix},\\ &= (1-s)(\mathsf{z}+(1-s))(\mathsf{z}^2-(1-s)^2)^{W-1}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}}), \qquad r\in[\![1,W]\!].\end{aligned}$$

The analogous computation for the odd moments yields

$$\begin{aligned}\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})_{1,2r+1} &= (\mathsf{z}^2-(1-s)^2)^{W-1}\det\begin{bmatrix}(s-1)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}}) & (1-s)(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})-1)\\ \mathsf{z}(s-1)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}}) & (1-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\end{bmatrix}\\ &= (s-1)(\mathsf{z}+(1-s))(\mathsf{z}^2-(1-s)^2)^{W-1}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})-1), \qquad r\in[\![1,W]\!].\end{aligned}$$

Some algebra provides, for $r\in[\![1,W]\!]$

$$(\mathbf{B}\boldsymbol{\epsilon})_p = \begin{cases} s\sum_{b=1}^{W}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2b}})\epsilon_{2b} + (2-s)\sum_{b=1}^{b=W}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2b}})-1)\epsilon_{2b+1}, & p=1,\\ s\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\epsilon_{2r} + (2-s)\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\epsilon_{2r+1}, & p=2r,\\ s\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\epsilon_{2r} + (2-s)\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})\epsilon_{2r+1}, & p=2r+1,\end{cases}$$

and thus, after tedious computations

$$\boldsymbol{e}_1^{\mathsf{t}}\mathrm{adj}(\mathsf{z}\boldsymbol{I}-\mathbf{A})\mathbf{B}\boldsymbol{\epsilon} = \mathsf{z}(\mathsf{z}+(1-s))(\mathsf{z}^2-(1-s)^2)^{W-1}\Big(s\sum_{r=1}^{W}\mathsf{A}(\mathsf{t}_{\boldsymbol{c}_{2r}})\epsilon_{2r} + (2-s)\sum_{r=1}^{W}(\mathsf{S}(\mathsf{t}_{\boldsymbol{c}_{2r}})-1)\epsilon_{2r+1}\Big).$$

To finish up, since the matrix $\mathsf{z}\boldsymbol{I}-\mathbf{A}$ is upper block triangular, we have

$$\det(\mathsf{z}\boldsymbol{I}-\mathbf{A}) = (\mathsf{z}-1)\prod_{r=1}^{W}\det(\mathsf{z}\boldsymbol{I}-\tilde{\mathbf{T}}_r\mathrm{diag}(1-s,s-1)) = (\mathsf{z}-1)(\mathsf{z}^2-(1-s)^2)^W,$$

giving the characteristic polynomial of the scheme. The property of $\Psi_2(\mathsf{z})$ annihilating for the first row of $\mathbf{E}$ can be checked analogously to Section 7.6.3. Observe that $\Psi_2(\mathsf{z})$ could also be found solving (10.53) by hand. □

10.4.3.3 Modified equations under acoustic scaling

The discussion of Section 10.4.3.2 is fully discrete. Now we come back to the asymptotic analysis using modified equations of Section 10.2 and considering local initialisations, *i.e.* $\mathbf{w}\in\mathbb{R}^q$.

Proposition 10.4.3: Modified equations

Under acoustic scaling, that is, when $\lambda>0$ is fixed as $\Delta x\to 0$, the modified equation for the bulk Finite Difference scheme (10.56) where the lattice Boltzmann scheme is determined by the choices of Example 7.6.2 with (10.61) is

$$\begin{aligned}&\partial_t\phi(t,\boldsymbol{x}) + \lambda\sum_{r=1}^{W}\epsilon_{2r}\sum_{|\mathbf{n}|=1}\boldsymbol{c}_{2r}^{\mathbf{n}}\partial_{\boldsymbol{x}}^{\mathbf{n}}\phi(t,\boldsymbol{x})\\ &\quad -\lambda\Delta x\Big(\frac{1}{s}-\frac{1}{2}\Big)\Big(2\sum_{r=1}^{W}\epsilon_{2r+1}\sum_{|\mathbf{n}|=2}\frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!}\partial_{\boldsymbol{x}}^{\mathbf{n}} - \Big(\sum_{r=1}^{W}\epsilon_{2r}\sum_{|\mathbf{n}|=1}\boldsymbol{c}_{2r}^{\mathbf{n}}\partial_{\boldsymbol{x}}^{\mathbf{n}}\Big)^2\Big)\phi(t,\boldsymbol{x}) = O(\Delta x^2),\end{aligned}$$

for $(t,\boldsymbol{x}) \in \mathbb{R}_+ \times \mathbb{R}^d$. Under the assumption of local initialisation $\mathbf{w} \in \mathbb{R}^q$ fulfilling Corollary 10.2.1, thus having $\mathsf{w}_1 = 1$ and $\mathsf{w}_{2r} = \epsilon_{2r}$ for $r \in [\![1, W]\!]$, the modified equation for the unique initialisation scheme ((10.55) with $n = 1$) is, for $\boldsymbol{x} \in \mathbb{R}^d$

$$\partial_t \phi(0,\boldsymbol{x}) + \lambda \sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \phi(0,\boldsymbol{x}) + O(\Delta x^2)$$
$$- \frac{\lambda \Delta x}{2} \Big(2 \sum_{r=1}^{W} \big((2-s)\epsilon_{2r+1} + (s-1)\mathsf{w}_{2r+1} \big) \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} - \Big(\sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \Big)^2 \Big) \phi(0,\boldsymbol{x}) = 0.$$

Proof. We have

$$\mathsf{A}(\mathbf{x}^{\boldsymbol{c}_{2r}}) \asymp -\Delta x \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} + O(\Delta x^3), \qquad \mathsf{S}(\mathbf{x}^{\boldsymbol{c}_{2r}}) \asymp 1 + \Delta x^2 \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} + O(\Delta x^4).$$

It can be easily checked that the modified equation of the bulk Finite Difference scheme reads as in the claim. For the initialisation scheme, only the computation of $\boldsymbol{\mathcal{E}}^{(0)}$, $\boldsymbol{\mathcal{E}}^{(1)}$ and $\boldsymbol{\mathcal{E}}^{(2)}$ is needed:

$$\boldsymbol{\mathcal{E}}_{1j}^{(0)} = \delta_{1j}, \quad \boldsymbol{\mathcal{E}}_{1,\cdot}^{(1)} = \Big(-s \sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}}, (s-1) \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}}, 0, \dots, (s-1) \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2W}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}}, 0 \Big),$$
$$\boldsymbol{\mathcal{E}}_{1,\cdot}^{(2)} = \Big((2-s) \sum_{r=1}^{W} \epsilon_{2r+1} \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}}, 0, (s-1) \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}}, \dots, 0, (s-1) \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2W}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} \Big).$$

Using the assumptions on the choice of initialisation, the modified equation for the initialisation scheme, which reads for $\boldsymbol{x} \in \mathbb{R}^d$

$$\partial_t \phi(0,\boldsymbol{x}) + \lambda \sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \phi(0,\boldsymbol{x})$$
$$- \frac{\lambda \Delta x}{2} \Big(2 \sum_{r=1}^{W} \big((2-s)\epsilon_{2r+1} + (s-1)\mathsf{w}_{2r+1} \big) \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} - \Big(\sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \Big)^2 \Big) \phi(0,\boldsymbol{x}) = O(\Delta x^2).$$

depends on the choice of initialisation w_{2r+1} of the odd moments, which still need to be fixed. □

Enforcing the equality between the dissipation coefficients of the initialisation scheme and the bulk Finite Difference scheme according to Proposition 10.4.3 provides the differential constraint

$$\sum_{r=1}^{W} \mathsf{w}_{2r+1} \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} = \frac{1}{s} \Big(\Big(\sum_{r=1}^{W} \epsilon_{2r} \sum_{|\mathbf{n}|=1} \boldsymbol{c}_{2r}^{\mathbf{n}} \partial_{\boldsymbol{x}}^{\mathbf{n}} \Big)^2 + (s-2) \sum_{r=1}^{W} \epsilon_{2r+1} \sum_{|\mathbf{n}|=2} \frac{\boldsymbol{c}_{2r}^{\mathbf{n}}}{\mathbf{n}!} \partial_{\boldsymbol{x}}^{\mathbf{n}} \Big).$$

We now provide some examples where this differential constraint can or cannot be fulfilled.

Example 10.4.3.

- $\mathrm{D_1Q_3}$, *having* $d = 1$, $W = 1$ *and* $c_2 = 1$. *After simplifying the second-order derivative operator, the condition reads* $\mathsf{w}_3 = (2\epsilon_2^2 + (s-2)\epsilon_3)/s$, *which has to be compared with* (10.52).
- $\mathrm{D_2Q_5}$, *having* $d = 2$, $W = 2$, $\boldsymbol{c}_2 = (1,0)^{\mathsf{t}}$ *and* $\boldsymbol{c}_4 = (0,1)^{\mathsf{t}}$, *we obtain*

$$\mathsf{w}_3 \partial_{x_1 x_1} + \mathsf{w}_5 \partial_{x_2 x_2} = \frac{1}{s} \big((2\epsilon_2^2 + (s-2)\epsilon_3) \partial_{x_1 x_1} + 4\epsilon_2 \epsilon_4 \partial_{x_1 x_2} + (2\epsilon_4^2 + (s-2)\epsilon_5) \partial_{x_2 x_2} \big),$$

  which cannot be fulfilled—except when either ϵ_2 *or* ϵ_4 *are zero rendering an essentially* 1d *problem—due to the presence of the mixed term in* $\partial_{x_1 x_2}$ *on the right hand side, arising from the hyperbolic part. In order to deal with this term, one is compelled to consider a richer scheme with diagonal discrete velocities, such as the* $\mathrm{D_2Q_9}$ *scheme.*
- $\mathrm{D_2Q_9}$, *having* $d = 2$, $W = 4$, $\boldsymbol{c}_2 = (1,0)^{\mathsf{t}}$, $\boldsymbol{c}_4 = (0,1)^{\mathsf{t}}$, $\boldsymbol{c}_6 = (1,1)^{\mathsf{t}}$ *and* $\boldsymbol{c}_8 = (-1,1)^{\mathsf{t}}$, *we obtain*

$$\frac{1}{2}(\mathsf{w}_3 + \mathsf{w}_7 + \mathsf{w}_9) \partial_{x_1 x_1} + (\mathsf{w}_7 - \mathsf{w}_9) \partial_{x_1 x_2} + \frac{1}{2}(\mathsf{w}_5 + \mathsf{w}_7 + \mathsf{w}_9) \partial_{x_2 x_2}$$

$$= \frac{1}{s}\Big(\underbrace{\Big(\epsilon_2^2+\epsilon_6^2+\epsilon_8^2+\epsilon_2\epsilon_6-\epsilon_2\epsilon_8-\epsilon_6\epsilon_8+\frac{1}{2}(s-2)(\epsilon_3+\epsilon_7+\epsilon_9)\Big)}_{R_{x_1x_1}} \partial_{x_1x_1}$$
$$+ \underbrace{\Big(2\epsilon_6^2-2\epsilon_8^2+2\epsilon_2\epsilon_4+\epsilon_2\epsilon_6+\epsilon_2\epsilon_8+\epsilon_4\epsilon_6-\epsilon_4\epsilon_8+(s-2)(\epsilon_7-\epsilon_9)\Big)}_{R_{x_1x_2}} \partial_{x_1x_2}$$
$$+ \underbrace{\Big(\epsilon_4^2+\epsilon_6^2+\epsilon_8^2+\epsilon_4\epsilon_6+\epsilon_4\epsilon_8+\epsilon_6\epsilon_8+\frac{1}{2}(s-2)(\epsilon_5+\epsilon_7+\epsilon_9)\Big)}_{R_{x_2x_2}} \partial_{x_2x_2}\Big).$$

This system is under-determined, thus it has several solutions. For example, picking $\mathsf{w}_9 = 0$*, we necessarily enforce* $\mathsf{w}_7 = R_{x_1x_2}/s$ *and then we have that* $\mathsf{w}_3 = (2R_{x_1x_1} - R_{x_1x_2})/s$ *and* $\mathsf{w}_5 = (2R_{x_2x_2} - R_{x_1x_2})/s$*.*

10.4.3.4 Modified equations under diffusive scaling

As previously analyzed, the literature also features lattice Boltzmann schemes used under diffusive scaling between time and space discretisations. We therefore finally consider this scaling where $\Delta t \propto \Delta x^2$, allowing to approximate the solution of

$$\begin{cases} \partial_t u(t,\boldsymbol{x}) + \boldsymbol{V}\cdot\nabla_{\boldsymbol{x}} u(t,\boldsymbol{x}) - \nabla_{\boldsymbol{x}}\cdot(\boldsymbol{D}\nabla_{\boldsymbol{x}} u)(t,\boldsymbol{x}) = 0, & (t,\boldsymbol{x}) \in \mathbb{R}_+ \times \mathbb{R}^d, \quad (10.65) \\ u(0,\boldsymbol{x}) = u^\circ(\boldsymbol{x}), & \boldsymbol{x} \in \mathbb{R}^d, \quad (10.66) \end{cases}$$

where the diffusion matrix is $\boldsymbol{D} \in \mathcal{M}_d(\mathbb{R})$. This scaling is difficult to treat in full generality because it requires a consistency study up to order $O(\Delta t) = O(\Delta x^2)$ included. Still, as previously highlighted, the unobservable framework of the current Section 10.4 allows us to circumvent these difficulties. The assumptions are slightly different than the rest of the paper.

Proposition 10.4.4: Modified equation of the bulk scheme

Under diffusive scaling, that is, when $\lambda = \mu/\Delta x$ with $\mu > 0$ fixed as $\Delta x \to 0$, assuming that $\epsilon_{2r} = \Delta x \hat{\epsilon}_{2r}$ where $\hat{\epsilon}_{2r}$ and ϵ_{2r+1} are fixed as $\Delta x \to 0$ for $r \in [\![1, W]\!]$, the modified equation for the bulk Finite Difference scheme (10.56) where the lattice Boltzmann scheme is determined by the choices of Example 7.6.2 with (10.61) is

$$\partial_t\phi(t,\boldsymbol{x}) + \mu\sum_{r=1}^{W}\hat{\epsilon}_{2r}\sum_{|\mathfrak{n}|=1}\boldsymbol{c}_{2r}^{\mathfrak{n}}\partial_{\boldsymbol{x}}^{\mathfrak{n}}\phi(t,\boldsymbol{x}) - 2\mu\Big(\frac{1}{s}-\frac{1}{2}\Big)\sum_{r=1}^{W}\epsilon_{2r+1}\sum_{|\mathfrak{n}|=2}\frac{\boldsymbol{c}_{2r}^{\mathfrak{n}}}{\mathfrak{n}!}\partial_{\boldsymbol{x}}^{\mathfrak{n}}\phi(t,\boldsymbol{x}) = O(\Delta x^2),$$

for $(t,\boldsymbol{x}) \in \mathbb{R}_+ \times \mathbb{R}^d$.

However, we observe that since $\Delta t \propto \Delta x^2$, the second-order consistency of the bulk scheme is preserved even when the initialisation schemes are not consistent, provided that $\mathsf{w}_1 = 1$. This is radically different from the acoustic scaling $\Delta t \propto \Delta x$ and comes from the fact that the errors coming from the initialisation routine are now of order $O(\Delta t) = O(\Delta x^2)$. Hence, under diffusive scaling, enforcing that the initialisation schemes are consistent is merely a question of obtaining time smoothness of the numerical solution.

Proposition 10.4.5

Under diffusive scaling, that is, when $\lambda = \mu/\Delta x$ with $\mu > 0$ fixed as $\Delta x \to 0$, assuming that $\epsilon_{2r} = \Delta x \hat{\epsilon}_{2r}$ where $\hat{\epsilon}_{2r}$ and ϵ_{2r+1} are fixed as $\Delta x \to 0$ for $r \in [\![1, W]\!]$, the modified equation of the unique initialisation scheme for the lattice Boltzmann scheme determined by the choices of Example 7.6.2 with (10.61)—considering a local initialisation $\mathbf{w} \in \mathbb{R}^q$ with $\mathsf{w}_1 = 1$—is

$$\partial_t\phi(0,\boldsymbol{x}) + \mu\sum_{r=1}^{W}(s\hat{\epsilon}_{2r} + (1-s)\hat{\mathsf{w}}_{2r})\sum_{|\mathfrak{n}|=1}\boldsymbol{c}_{2r}^{\mathfrak{n}}\partial_{\boldsymbol{x}}^{\mathfrak{n}}\phi(0,\boldsymbol{x})$$
$$-\mu\sum_{r=1}^{W}((2-s)\epsilon_{2r+1} + (s-1)\mathsf{w}_{2r+1})\sum_{|\mathfrak{n}|=2}\frac{\boldsymbol{c}_{2r}^{\mathfrak{n}}}{\mathfrak{n}!}\partial_{\boldsymbol{x}}^{\mathfrak{n}}\phi(0,\boldsymbol{x}) = O(\Delta x),$$

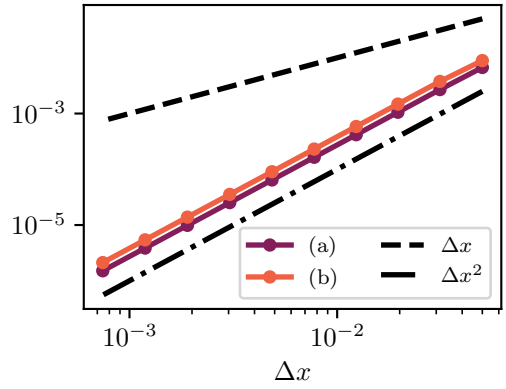


Figure 10.12: L^2 errors at the final time for the initialisations (a) and (b). We observe second-order convergence irrespective of the consistency of the initialisation scheme.

for $\boldsymbol{x} \in \mathbb{R}^d$, where $\mathsf{w}_{2r} = \Delta x \hat{\mathsf{w}}_{2r}$ with fixed $\hat{\mathsf{w}}_{2r}$ as $\Delta x \to 0$ for $r \in [\![1, W]\!]$.

Therefore, the initialisation scheme is consistent with the bulk scheme under the conditions

$$\hat{\mathsf{w}}_{2r} = \hat{\epsilon}_{2r}, \qquad \mathsf{w}_{2r+1} = \frac{s-2}{s}\epsilon_{2r+1}, \qquad r \in [\![1, W]\!],$$

which are only set to ensure—as previously stated—time smoothness.

To numerically verify the previous claims, we consider the $\mathrm{D}_1\mathrm{Q}_3$ introduced in Example 10.4.3. Considering the bounded domain $\Omega = [0,1]$ with periodic boundary conditions, using $u^\circ(x) = \cos(2\pi x)$ renders the exact solution $u(t,x) = e^{-4\pi^2 D t}\cos(2\pi(x - Vt))$ to (10.65)/(10.66). We utilize $\mu = 1$, $V = 2$ and $D = 1/32$. These physical constant are set taking $\hat{\epsilon}_2 = V$, $\epsilon_3 = 1$ and $s = 1/(D + 1/2)$. We consider two kinds of initialisations, which are

$$\text{(a)} \qquad \mathsf{w}_1 = 1, \quad \mathsf{w}_2 = \Delta x \tilde{\epsilon}_2, \quad \mathsf{w}_3 = \frac{s-2}{s}\epsilon_3,$$
$$\text{(b)} \qquad \mathsf{w}_1 = 1, \quad \mathsf{w}_2 = \frac{\Delta x \tilde{\epsilon}_2}{2}, \quad \mathsf{w}_3 = 10\frac{s-2}{s}\epsilon_3,$$

with the first condition (a) yielding a consistent initialisation scheme and the second condition (b) giving an inconsistent one.

10.4.3.4.1 Study of the convergence order We simulate until the final time $T = 0.05$ and measure the L^2 errors progressively decreasing the space step Δx. The results are given in Figure 10.12, confirming that, regardless of the consistency of the initialisation scheme, the overall method is second-order convergent, since $\Delta t \propto \Delta x^2$. As expected, the error constant is slightly better when the initialisation scheme is consistent.

10.4.3.4.2 Study of the time smoothness of the numerical solution We consider a framework analogous to the one of Section 10.3.1.2 with $\Delta x = 1/30$ and the previously introduced parameters, measuring the discrepancy between numerical and exact solution at the eighth point of the lattice. The results in Figure 10.13 confirm the previous theoretical discussion: considering consistent initialisation schemes allows to avoid initial oscillating boundary layers in the simulation. Furthermore, we see that for the even steps, the transport and diffusion coefficients from the modified equations of the starting schemes are always closer to the one in the bulk than the ones for the odd steps, explaining why the discrepancies in terms of error with respect to the exact solution are smaller for even steps than for odd steps.

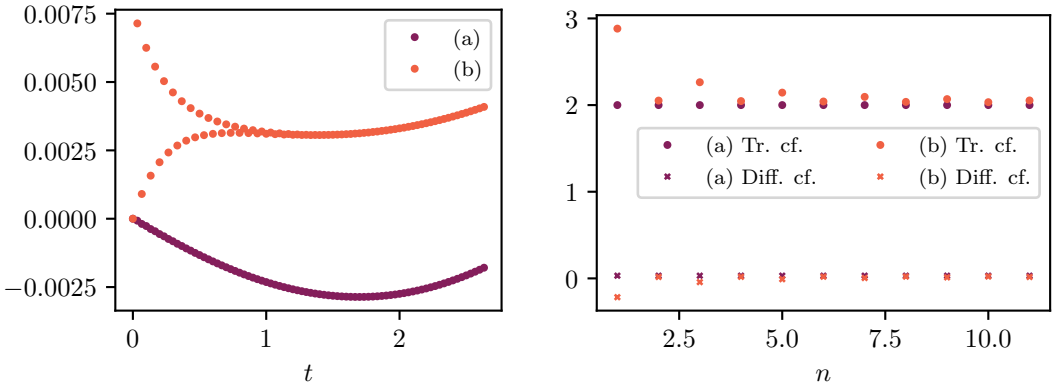


Figure 10.13: Left: test for smoothness in time close to $t = 0$ for the initialisations (a) and (b): difference between exact and numerical solution at the eighth lattice point. The first one gives a smooth profile whereas the second one oscillates with damping. Right: transport and diffusion coefficients in the modified equations for different n.

10.4.4 Conclusions

In Section 10.4, we have defined a notion of observability for lattice Boltzmann schemes, allowing to identify the schemes with fewer initialisation schemes than those prescribed by Q (the number of non-conserved moment not relaxing on the equilibrium) as those being unobservable. In control theory, it is well known that unobservable systems can be represented by other systems which order has been reduced removing unobservable modes. It is therefore easy to analyze the initialisation phase of these schemes with the technique of the modified equation. In particular, we have found that a well-known and vast class of lattice Boltzmann schemes, namely the so-called link two-relaxation-times schemes with magic parameters equal to one-fourth [Ginzburg, 2009, d'Humières and Ginzburg, 2009], fits this framework. We have exploited this fact in order to provide the constraints on the initial data for having a smooth initialisation, both under acoustic and diffusive scaling.

10.5 Conclusions of Chapter 10

Due to the fact that lattice Boltzmann schemes feature more unknowns than variables of interest, their initialisation—especially for the non-conserved moments—can have an important impact on the outcomes of the simulations. The aim of Chapter 10 was indeed to study the role of the initialisation on the numerical behaviour of general lattice Boltzmann schemes. To this end, we have introduced a modified equation analysis which has ensured to propose initialisations yielding consistent initialisation schemes, which is crucial to preserve the second-order accuracy of many schemes. The modified equation has also allowed to precisely describe the behaviour of the lattice Boltzmann schemes close to the beginning of the numerical simulation—where the different dynamics were essentially driven by numerical dissipation—and identify initialisations yielding smooth numerical solutions without oscillatory initial layers. Finally, we have introduced a notion of observability for lattice Boltzmann schemes which has allowed to characterize schemes with a small number of initialisation schemes. This feature makes the study of the initialisation for these schemes way more accessible than for general ones. Consistent and smooth initialisations have been a hot topic in the lattice Boltzmann community for quite a long time [Van Leemput et al., 2009, Caiazzo, 2005, Junk and Yang, 2015, Huang et al., 2015b]. However, to the best of our knowledge, no general approach to the analysis of these features were available. They are important in order to ensure that the order of the schemes is preserved. From another perspective, although the novel notion of observability for lattice Boltzmann schemes has been exploited solely to study the number of needed initialisation schemes, we do believe that it can be useful to investigate other features of these schemes. For example, one interesting topic would be the one linked to "realisation" [Brewer et al., 1986, Chapter 4] and "minimal realisations" [De Schutter, 2000]: given a target Finite Difference scheme (*i.e.* a transfer function), how can we construct the smallest lattice Boltzmann

scheme of which it is the corresponding Finite Difference scheme. This will be the object of future investigations.

Summary and perspectives of Part III

In Chapter 7, we have first eliminated the non-conserved moments, obtaining multi-step Finite Difference schemes only on the variables of interest. We have therefore observed that the natural notions of consistency and stability for lattice Boltzmann schemes are those which are already known for Finite Difference schemes. Still—having observed that obtaining the corresponding Finite Difference scheme can be complicated—we have studied these properties directly on the lattice Boltzmann scheme without explicit computations. To this end, in Chapter 8, we have proved results on the consistency of general lattice Boltzmann schemes without explicitly computing the corresponding Finite Difference scheme. Moreover, in Chapter 9, we have shown that the *von Neumann* stability analysis which is currently employed on lattice Boltzmann schemes entirely meets the one on Finite Difference schemes. In order to fully conclude on the convergence of consistent and stable lattice Boltzmann schemes, the initialisation must be correctly taken into account, which is the subject of Chapter 10. Thus, we have proposed an analysis of the initialisation process, based on the modified equation, for linear schemes with one conserved moment. This has allowed to find the constraint to have consistent initialisation schemes and has also precisely characterized the smoothness of the numerical solution in terms of compatible dissipation between initialisation and bulk schemes.

Two main tracks for future research—which are currently works in progress—have been left aside and are the subject of Part IV. They are:

1. Use the corresponding Finite Difference scheme to clarify the issue of stability of lattice Boltzmann with respect to other norms. One interesting type of stability is the one for non-linear schemes in the L^∞, linked with monotonicity and maximum principles [Dellacherie, 2014, Caetano et al., 2023]. This can be first investigated on specific simple schemes and then—hopefully—on general lattice Boltzmann schemes. Weighted L^2 norms—for which several works are available [Banda et al., 2006, Junk and Yong, 2009, Rheinländer, 2010]—will also be of interest for future investigations.
2. Use the corresponding Finite Difference scheme to clarify the issue of consistency and stability for boundary conditions. Our way of recasting the scheme only on the conserved moments cannot work as it is, since the presence of a boundary introduces a lack of spatial invariance of the scheme under spatial translations. This prevents us from utilizing a general theorem such as the Cayley-Hamilton theorem. The plan is to start by overcoming these difficulties on simple lattice Boltzmann schemes and then try to look for a general strategy.

Part IV

Perspectives on numerical analysis of lattice Boltzmann schemes

Aim and structure of Part IV

The aim of Part IV is to gather two preliminary and unaccomplished studies on issues that have been raised in the conclusions of Part III. The first one is the study of the stability of a simple non-linear lattice Boltzmann scheme for the L^∞ norm, which allows us to study its convergence towards the weak solution of the target equation. This is the topic of Chapter 11. A second topic is the study of boundary conditions, which is developed in Chapter 12.

Chapter 11

Convergence of the D_1Q_2 scheme towards the weak solution of a scalar conservation law

General context and motivation

In the conclusions of Part III, we have stated that other kinds of stability besides the linear L^2 one are interesting. In particular, as far as non-linear schemes are concerned, the L^∞ stability, linked with monotonicity and maximum principles, is one of the leading tools to prove convergence in the case of Finite Difference/Finite Volume methods towards weak entropic solutions.

State of the art

To the best of our knowledge, this topic has only been investigated in two contributions, concerning to the D_1Q_2 scheme. The first one is [Dellacherie, 2014], who limits the study to a linear framework and utilizes different approaches according to the value of the relaxation parameter. The author relies either on the corresponding Finite Difference scheme or on a different "pseudo" lattice Boltzmann scheme (called "LBM*"). A second work is the one by [Caetano et al., 2023]. Here, the study is conducted on the original lattice Boltzmann scheme by using the fact that the stream phase is a mere shift of the data and that—for a relaxation parameter smaller or equal to one—the collision phase is a convex combination. Then, convergence towards the weak entropic solution of a scalar conservation law is proved, upon extraction, in a classical fashion, relying on the quasi-equilibrium of the non-conserved moment. The limitation of this work lies in the fact that it treats the under-relaxation regime only.

Aims and structure of Chapter 11

The aim of Chapter 11 is to extend and—if possible—encompass the work of [Caetano et al., 2023] to prove the convergence of the D_1Q_2 towards the weak entropic solution of a scalar conservation law, in particular in the over-relaxation regime. However, we want to do this relying on the corresponding Finite Difference scheme, forgetting—as long as it is possible—that it comes from a specific underlying lattice Boltzmann scheme. This choice is dictated by the fact that—upon taking the initialization into account (*cf.* Chapter 10)—the corresponding Finite Difference contains the whole discrete dynamics of the original lattice Boltzmann scheme. Moreover, the generality of the corresponding Finite Difference scheme is promising to extend the study to more complex schemes. Chapter 11 is structured as follows. In Section 11.1, we recall the basic theory for the existence and uniqueness of the solution to a scalar conservation law for $d = 1$ and the tools concerning bounded variation which shall be needed. In Section 11.2, the D_1Q_2 scheme and its corresponding Finite Difference scheme are recalled, with particular care devoted to the choice of initial datum and the initialization of the scheme, for they play an important role in the discussion on stability. We also emphasize the difference between the under-relaxation (relaxation parameter smaller than one) and the over-relaxation (relaxation parameter larger than one) regimes. The difference between the two is studied further in Section 11.3, which is the last part of Chapter 11 where we

consider the original lattice Boltzmann scheme. Here, we also discuss monotonicity, a maximum principle, and L^∞ stability for the corresponding Finite Difference scheme under linearity assumption. The core Section 11.4 is devoted to investigate the non-linear framework in the over-relaxation regime and demonstrate—by passing through the corresponding Finite Difference scheme—the convergence of the original lattice Boltzmann scheme. This essentially relies on the path by [Caetano et al., 2023]. Section 11.5 tries to show that the maximum principle holds also in the under-relaxation regime (as proved by [Caetano et al., 2023] on the original lattice Boltzmann scheme), respecting the constraint of working uniquely on the corresponding Finite Difference scheme. This is first done in a linear setting to fix ideas and extended to the non-linear framework in Section 11.6. Unfortunately, in the path to recover the result by [Caetano et al., 2023] using the standpoint of the corresponding Finite Difference scheme, we are left with unsolved issues in the proof to be conducted in the spirit of Section 11.4. These points are presented and discussed in Section 11.7 for future investigations.

Contents

11.1 Continuous problem

11.1.1 Strong and weak formulation

The target problem we are interested in is the same as (2.50). We consider a non-linear scalar conservation law under the form of the Cauchy problem

$$\begin{cases} \partial_t u(t,x) + \partial_x(\varphi(u))(t,x) = 0, & t \in [0,T], \quad x \in \mathbb{R}, \\ u(0,x) = u^\circ(x), & x \in \mathbb{R}, \end{cases} \tag{11.1}$$

where we assume that the flux is $\varphi \in C^1(\mathbb{R})$. Concerning the smoothness of the initial datum, in order to ensure the needed existence and uniqueness theorems for the forthcoming weak formulation, selecting $u^\circ \in L^\infty(\mathbb{R})$ is enough [Harten, 1984, Godlewski and Raviart, 1991, Godlewski and Raviart, 1996, Eymard et al., 2000, Caetano et al., 2023]. Still, according to the previous references, since we shall need total variation estimates from the total variation of the initial datum, we enforce $u^\circ \in L^\infty(\mathbb{R}) \cap \mathrm{BV}(\mathbb{R}) \equiv \mathrm{BV}(\mathbb{R})$. Indeed, in the monodimensional case—see [Godlewski and Raviart, 1991, Page 159] or [Eymard et al., 2000, page 872]—$\mathrm{BV}(\mathbb{R}) \subset L^\infty(\mathbb{R})$.

Remark 11.1.1. *If the solution of (11.1) is sought—for some reason—in an open set $K \subset \mathbb{R}$, to which it has to belong almost everywhere in time and space, the following theory remains essentially unchanged and can be easily adapted. In this framework, one considers $\varphi \in C^1(K)$, $u^\circ \in K$ almost everywhere, $u \in K$ almost everywhere, and when defining entropy-entropy flux functions, their domain of definition will be K with the assumption that K is convex.*

Since we do not restrain the research to classical solutions, we need a suitable notion of weak solution. The following definition is the one given by [Godlewski and Raviart, 1991, Godlewski and Raviart, 1996, Eymard et al., 2000].

Definition 11.1.1: Weak solution

We say that $u : \mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$ is a weak solution of (11.1) if $u \in L^\infty(\mathbb{R}_+ \times \mathbb{R})$ and

$$\int_0^{+\infty} \int_{\mathbb{R}} \big(u(t,x)\partial_t \phi(t,x) + \varphi(u(t,x))\partial_x \phi(t,x)\big)\,\mathrm{d}x\mathrm{d}t + \int_{\mathbb{R}} u^\circ(x)\phi(0,x)\mathrm{d}x = 0, \qquad \forall \phi \in C_c^1(\mathbb{R}_+ \times \mathbb{R}).$$

Remark 11.1.2 (Smoothness of the test function)**.** *Slightly different weak formulations, where the test functions belong to $C_c^\infty(\mathbb{R}_+ \times \mathbb{R})$, are proposed in [Harten et al., 1976, Harten and Lax, 1981, Harten, 1984, Serre, 1999, Caetano et al., 2023]. This additional smoothness is not necessary, see [Ambrosio et al., 2000, Page 117]. Some formulations [Sanders, 1983] do not care about the initial conditions, thus take $\phi \in C_c^\infty(\mathbb{R}_+^\star \times \mathbb{R})$.*

11.1.2 Entropic weak solution

In order to obtain a uniqueness result and to select the physical weak solution, one utilizes the concept of mathematical entropy. We consider the definition by [Godlewski and Raviart, 1991, Godlewski and Raviart, 1996].

Definition 11.1.2: Entropy-entropy flux pair

A convex function $\eta : \mathbb{R} \to \mathbb{R}$ is an entropy for (11.1) if there exists $q : \mathbb{R} \to \mathbb{R}$, called entropy flux, such that $\eta'(u)\varphi'(u) = q(u)$ for every $u \in \mathbb{R}$.

Therefore, we have that:

Definition 11.1.3: Entropy weak solution

A weak solution according to Definition 11.1.1 is said to be an entropy weak solution if for any entropy function from Definition 11.1.2, it satisfies

$$\int_0^{+\infty} \int_{\mathbb{R}} \big(\eta(u(t,x))\partial_t \phi(t,x) + q(u(t,x))\partial_x \phi(t,x)\big)\,\mathrm{d}x\mathrm{d}t + \int_{\mathbb{R}} \eta(u^\circ(x))\phi(0,x)\mathrm{d}x \geq 0,$$
$$\forall \phi \in C_c^1(\mathbb{R}_+ \times \mathbb{R}), \quad \phi \geq 0.$$

In practice, contrarily to [Caetano et al., 2023], we shall make use of a particular choice of family of entropies, called Krushkov entropies. They come under the form, for any $\kappa \in \mathbb{R}$

$$\eta(u) = |u - \kappa|, \qquad q(u) = \mathrm{sign}(u - \kappa)(\varphi(u) - \varphi(\kappa)) = \varphi(u \top \kappa) - \varphi(u \bot \kappa),$$

where the notation $a \top b := \max(a,b)$ and $a \bot b := \min(a,b)$ for any $a, b \in \mathbb{R}$ is used. The discussion in [Godlewski and Raviart, 1991, Page 72 and 73] and [Serre, 1999, Page 35], recalled in [Eymard et al., 2000], concludes that Definition 11.1.3 is equivalent to the following utilizing Krushkov entropies:

Proposition 11.1.1: Entropy weak solution using Krushkov entropies

A solution is an entropy weak solution according to Definition 11.1.3 if and only if for every $\kappa \in \mathbb{R}$

$$\int_0^{+\infty}\int_{\mathbb{R}} \big(|u(t,x)-\kappa|\partial_t\phi(t,x) + (\varphi(u(t,x)\top\kappa)-\varphi(u(t,x)\bot\kappa)\partial_x\phi(t,x)\big)\,\mathrm{d}x\mathrm{d}t + \int_{\mathbb{R}} |u^\circ(x)-\kappa|\phi(0,x)\mathrm{d}x \geq 0, \qquad \forall \phi \in C_c^\infty(\mathbb{R}_+\times\mathbb{R}), \quad \phi \geq 0.$$

11.1.3 Existence and uniqueness of the weak entropic solution

Merging the results from [Godlewski and Raviart, 1991, Godlewski and Raviart, 1996, Serre, 1999] and [Eymard et al., 2000], we obtain

Theorem 11.1.1: Existence and uniqueness

Let $\varphi \in C^1(\mathbb{R})$ and $u^\circ \in L^\infty(\mathbb{R})$. Then there exists a unique entropy weak solution $u \in L^\infty(\mathbb{R}_+\times\mathbb{R}) \cap \mathcal{C}(\mathbb{R}_+; L^1_{\mathrm{loc}}(\mathbb{R}))$[a] to (11.1) according to Definition 11.1.3. Moreover

- $\|u(t,\cdot)\|_{L^\infty(\mathbb{R})} \leq \|u^\circ\|_{L^\infty(\mathbb{R})}$ for almost every $t \in \mathbb{R}_+$.
- If $u^\circ \in \mathrm{BV}(\mathbb{R})$, then $|u(t,\cdot)|_{\mathrm{BV}(\mathbb{R})} \leq |u^\circ|_{\mathrm{BV}(\mathbb{R})}$ for any $t \in \mathbb{R}_+$, thus $u(t,\cdot) \in \mathrm{BV}(\mathbb{R})$ for any $t \in \mathbb{R}_+$.

[a] Where $\mathcal{C}$ indicates continuous bounded functions.

11.1.4 Survival kit on bounded-variation theory

Until now, we have utilized the notion of space of bounded variation functions without precisely define it. Here, we provide the definition of this space as well as some important properties which shall be useful in the rest of Chapter 11. We give the definition of BV according to [Giusti and Williams, 1984, Godlewski and Raviart, 1991]

Definition 11.1.4: Bounded variation

Let $\Omega \subset \mathbb{R}^p$ for $p \geq 1$[a] be an open (not necessarily bounded) set. Let $u : \Omega \to \mathbb{R}$ be a function of class $u \in L^1_{\mathrm{loc}}(\Omega)$. We define

$$|u|_{\mathrm{BV}(\Omega)} := \sup\left\{\int_\Omega u \mathrm{div}\phi \quad : \quad \phi \in C_c^1(\Omega), \quad \|\phi\|_{L^\infty(\Omega)} \leq 1\right\}.$$

With this semi-norm, the space $\mathrm{BV}(\Omega)$ is defined as

$$\mathrm{BV}(\Omega) := \left\{u \in L^1_{\mathrm{loc}}(\Omega) \quad : \quad |u|_{\mathrm{BV}(\Omega)} < +\infty\right\}.$$

[a] We shall take $p = 2$ in this work, because we deal with time and space

Observe that we do not include the additional integrability requirement $L^1(\Omega)$ as [Giusti and Williams, 1984] in Definition 11.1.4. However, we have to require it now to state the following result, see [Godlewski and Raviart, 1991].

Proposition 11.1.2

The space $L^1(\Omega)\cap\mathrm{BV}(\Omega)$ is a Banach space for the norm

$$\|u\|_{L^1(\Omega)\cap\mathrm{BV}(\Omega)} := \|u\|_{L^1(\Omega)} + |u|_{\mathrm{BV}(\Omega)}, \qquad u \in L^1(\Omega)\cap\mathrm{BV}(\Omega).$$

The last result that we need for this work is a compactness theorem. For this, we restate and merge results from [Giusti and Williams, 1984, Godlewski and Raviart, 1991, Ambrosio et al., 2000]

Theorem 11.1.2

Let $\Omega \subset \mathbb{R}^p$ for $p \geq 1$ be a bounded open set with Lipschitz continuous boundary. Let $(u_k)_{k\in\mathbb{N}} \subset L^1(\Omega) \cap \mathrm{BV}(\Omega)$ be a norm bounded sequence for the norm of $L^1(\Omega)\cap \mathrm{BV}(\Omega)$, then there exists an extraction $\psi : \mathbb{N} \to \mathbb{N}$ and a limit $\overline{u} \in L^1(\Omega) \cap \mathrm{BV}(\Omega)$ such that

$$u_{\psi(k)} \to \overline{u} \quad \text{for the topology of } L^1(\Omega),$$
$$|\overline{u}|_{\mathrm{BV}(\Omega)} \leq \liminf_{k\to+\infty} |u_{\psi(k)}|_{\mathrm{BV}(\Omega)}.$$

Remark 11.1.3. *Observe that the fact that the sequence must be bounded in the $\|\cdot\|_{L^1(\Omega)\cap BV(\Omega)}$ norm requires to be more than $BV(\Omega)$ according to our definition based on $L^1_{\mathrm{loc}}(\Omega)$, namely also to be $L^1(\Omega)$. Moreover, the convergence of the extracted subsequence takes place in the $L^1(\Omega)$ but not in the $\|\cdot\|_{L^1(\Omega)\cap BV(\Omega)}$ topology, which would add the BV seminorm.*

11.2 Lattice Boltzmann scheme and corresponding Finite Difference scheme

11.2.1 Space and time discretizations

We assume to work on an unbounded lattice $\Delta t\mathbb{N} \times \Delta x\mathbb{Z}$. However, the notations shall be slightly different compared to Chapter 1 and Part III. The discrete points shall not be denoted $(t, x) \in \Delta t\mathbb{N} \times \Delta x\mathbb{Z}$ but we use $t^n := n\Delta t \in \Delta t\mathbb{N}$ for any $n \in \mathbb{N}$ and $x_j := j\Delta x \in \Delta x\mathbb{Z}$ for any $j \in \mathbb{Z}$. We use the so-called acoustic scaling corresponding to $\Delta x/\Delta t = \lambda$ kept fixed as $\Delta x \to 0$.

11.2.2 Lattice Boltzmann scheme

We study the $\mathrm{D}_1\mathrm{Q}_2$ lattice Boltzmann scheme introduced in Section 1.5.1. For the sake of clarity, we change the notations and we will indicate (left hand side: new notation, right hand side: old notation)

$$\mathsf{f}_j^{+,n} = \mathsf{f}_1(t^n, x_j), \qquad \mathsf{f}_j^{-,n} = \mathsf{f}_2(t^n, x_j), \qquad \mathsf{u}_j^n = \mathsf{m}_1(t^n, x_j), \qquad \mathsf{v}_j^n = \mathsf{m}_2(t^n, x_j).$$

Moreover, in order to have that $\mathsf{u}_j^n \approx u(t^n, x_j)$ (or its averages) from (11.1), we consider the equilibrium

$$m_2^{\mathrm{eq}}(\mathsf{m}_1) = \varphi(\mathsf{m}_1)$$

The relaxation parameter s_2 will be called s since no confusion is possible. The original lattice Boltzmann scheme reads:

- **Initialization**. We discretize the conserved moment u using the averages of the initial datum u°. The non-conserved moment is taken at its local equilibrium value. This reads

$$\mathsf{u}_j^0 = \frac{1}{\Delta x}\int_{x_j}^{x_{j+1}} u^\circ(x)\mathrm{d}x, \qquad \mathsf{v}_j^0 = \varphi(\mathsf{u}_j^0), \qquad j \in \mathbb{Z}. \tag{11.2}$$

  Remark 11.2.1 (On the choice of initialization)**.** *Observe that contrarily to Chapter 10, we do not initialize with point values of the initial datum but we take averages. Second, it is important to notice that the non-conserved moment is taken at the equilibrium,* cf. *Chapter 10. The importance of this choice shall be capital.*

- **Scheme** for $n \in \mathbb{N}$, split into two phases:
  - **Collision phase**. This phase is local in space and reads

$$\mathsf{u}_j^{n,\star} = \mathsf{u}_j^n, \qquad \mathsf{v}_j^{n,\star} = (1-s)\mathsf{v}_j^n + s\varphi(\mathsf{u}_j^n), \qquad j \in \mathbb{Z}, \tag{11.3}$$

where the relaxation parameter $s \in]0,2]$. This part of the algorithm is diagonal in the variables u and v. Then, the post-collision distribution densities are retrieved by using $\boldsymbol{M}^{-1}$

– **Stream phase**. This linear phase is not local to each site of the lattice and reads

$$\mathsf{f}_j^{\pm,n+1} = \mathsf{f}_{j\mp1}^{\pm,n,\star}, \qquad j \in \mathbb{Z}. \tag{11.4}$$

Observe that this part of the algorithm is diagonal in the variables f^+ and f^-. Then we recover u and v by applying $\boldsymbol{M}$.

11.2.3 Corresponding Finite Difference scheme

We have seen in Chapter 7 and Chapter 10, that the conserved moment computed by the lattice Boltzmann scheme and the one computed using the corresponding Finite Difference scheme are the same for each time step and each point on the spatial grid. The corresponding Finite Difference scheme reads:

- **Initialization**
$$\mathsf{u}_j^0 = \frac{1}{\Delta x}\int_{x_j}^{x_{j+1}} u^\circ(x)\mathrm{d}x, \qquad j \in \mathbb{Z}.$$

- **Initial scheme** for $n = 0$, which depends on the choice that we took for v^0, namely (11.2).
$$\mathsf{u}_j^1 = \frac{1}{2}(\mathsf{u}_{j-1}^0 + \mathsf{u}_{j+1}^0) + \frac{1}{2\lambda}\big(\varphi(\mathsf{u}_{j-1}^0) - \varphi(\mathsf{u}_{j+1}^0)\big), \qquad j \in \mathbb{Z}, \tag{11.5}$$
which corresponds to a Lax-Friedrichs discretization of the target equation (11.1), thus is consistent.

- **Bulk scheme** for $n \in \mathbb{N}^*$.
$$\mathsf{u}_j^{n+1} = \Big(1 - \frac{s}{2}\Big)(\mathsf{u}_{j-1}^n + \mathsf{u}_{j+1}^n) + \frac{s}{2\lambda}\big(\varphi(\mathsf{u}_{j-1}^n) - \varphi(\mathsf{u}_{j+1}^n)\big) + (s-1)\mathsf{u}_j^{n-1}, \qquad j \in \mathbb{Z}. \tag{11.6}$$

The corresponding Finite Difference scheme (11.6) is the combination of three well-known Finite Difference schemes, namely

- A scheme for the wave equation at velocities $\pm\lambda$ obtained when $s = 0$.
- The Lax-Friedrichs scheme—obtained when $s = 1$. This scheme is known for being total variation diminishing under the CFL condition.
- The leap-frog scheme—obtained when $s = 2$. This scheme is the prototype of non total variation diminishing scheme and generates eminently oscillatory behavior.

We can therefore identify two regimes, which names are inherited from the lattice Boltzmann point of view, namely

- The over-relaxation (OR), when $s \in]1,2]$, where (11.6) is a convex combination of the Lax-Friedrichs (weight $2-s$) and the leap-frog scheme (weight $s-1$). This is the area where much can be said using the corresponding Finite Difference scheme. To the best of our knowledge, this regime has never been studied in the non-linear framework. For the linear framework, refer to [Dellacherie, 2014].
- The under-relaxation (UR) for $s \in]0,1]$, where the weight for the leap-frog scheme is negative, meaning that we are adding "way more" Lax-Friedrichs. Still, the scheme is a convex combination of a scheme for the wave equation (weight $1-s$) at velocities $\pm\lambda$ and Lax-Friedrichs scheme (weight s). It is harder to study using the corresponding Finite Difference scheme as far as maximum principles are involved. This regime has been studied by [Caetano et al., 2023] on the original lattice Boltzmann formulation (11.2), (11.3) and (11.4).

11.3 Linear framework: under and over-relaxation

In order to further understand the differences between the OR and the UR regimes, it is useful to analyze the linear version of the lattice Boltzmann scheme and the corresponding Finite Difference scheme. This corresponds to taking the equilibrium of the second moment equal to $\varphi(u) = Vu$ for some transport velocity $V \in \mathbb{R}$.

11.3.1 Non negativity of the collision matrix

A first analysis which can be done pertains to the original lattice Boltzmann scheme, not the corresponding Finite Difference scheme. Still, keep in mind that our policy is to only use the Finite Difference scheme to study the lattice Boltzmann scheme. We can rewrite (11.3) on $\mathsf{f}^{\pm}$ instead of on u and v and have a collision matrix—*cf.* $\boldsymbol{K}$ from (10.3)—for $\mathsf{f}^{\pm}$. We do not list the space and time indices.

$$\begin{bmatrix} \mathsf{f}^{+,\star} \\ \mathsf{f}^{-,\star} \end{bmatrix} = \boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \begin{bmatrix} \mathsf{f}^{+} \\ \mathsf{f}^{-} \end{bmatrix}, \qquad \text{where} \qquad \boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} = \begin{bmatrix} 1-\frac{s}{2}\left(1-\frac{V}{\lambda}\right) & \frac{s}{2}\left(1+\frac{V}{\lambda}\right) \\ \frac{s}{2}\left(1-\frac{V}{\lambda}\right) & 1-\frac{s}{2}\left(1+\frac{V}{\lambda}\right) \end{bmatrix}.$$

The interesting quantity is the Courant number $C := V/\lambda$. Following the discussion by [Dubois et al., 2020a], one can consider the values of s and C where the entries of the matrix $\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}$ are non-negative. This guarantees that, since the transport phase (11.4) is a simple shift in the space of $\mathsf{f}^{\pm}$, if $\mathsf{f}_j^{\pm,0} \geq 0$ for every $j \in \mathbb{Z}$, eventually $\mathsf{f}_j^{\pm,n} \geq 0$ for every $n \in \mathbb{N}$ and $j \in \mathbb{Z}$.

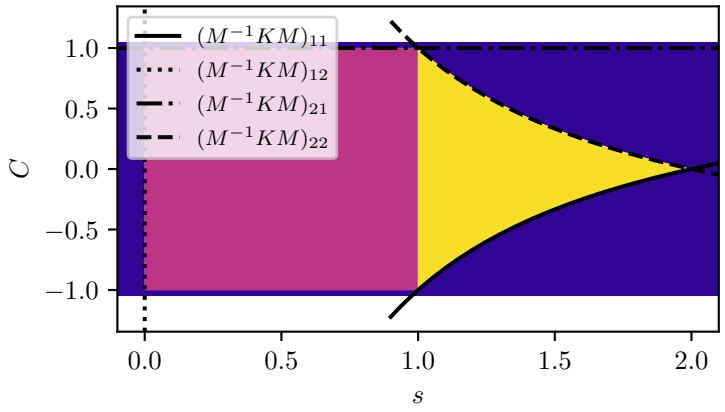


Figure 11.1: Study of the areas where $\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}$ is non-negative. The violet part concerns the UR regime and corresponds to $s \in]0,1]$ with $|C| \leq 1$. The yellow part pertains to the OR regime with $s \in [1,2]$ and $|C| \leq (2-s)/s$. The constraints associated with different matrix entries are highlighted using different styles of black lines.

Proposition 11.3.1: Non-negative collision

The matrix $\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}$ is non-negative, that is $(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{ij} \geq 0$ for $i, j \in [\![1,2]\!]$, if and only if the following conditions are met:

$$|C| \leq 1, \qquad \text{and} \qquad 0 \leq s \leq \frac{2}{1+|C|}.$$

These conditions are equivalent to

$$s \in [0,2], \qquad \text{and} \qquad |C| \leq \min\left(1, \frac{2-s}{s}\right), \qquad (\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0)$$

where ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0$) will be a shorthand for this "non-negative collision" condition.

Proof. Without loss of generality, select $C \geq 0$. Then

- We have $(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{12} \geq 0$ if and only if $s \geq 0$.
- Since by the previous point, $s \geq 0$, in order to have $(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{21} \geq 0$, we need the condition $C \leq 1$.
- Enforcing $(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{11} \geq 0$ taking the previous constraints into account yields $s \leq 2/(1-C)$.
- Finally, enforcing $(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{22} \geq 0$ gives $s \leq 2/(1+C)$.

We conclude by observing that if $0 \leq C \leq 1$, then $\min(2/(1-C), 2/(1+C)) = 2/(1+C)$. The equivalent condition comes from simple algebraic manipulations. □

The constraints of Proposition 11.3.1 are depicted in Figure 11.1. It is rather clear by looking at it that the lattice Boltzmann scheme behaves radically differently in the OR zone compared to the UR one.

Remark 11.3.1. *We observe that the positivity zone for the UR regime, namely the square $s \in [0,1]$ and $|C| \in [0,1]$ is the one where the results by [Caetano et al., 2023] are valid.*

Remark 11.3.2. *We remark that the condition $s \in [1,2]$ and $s = 2/(1+|C|)$ is the one that is used in [Dellacherie, 2014, Proposition 8.1] to state a maximum principle on the Finite Difference scheme.*

11.3.2 Monotonicity, maximum principle and L^∞ stability of the corresponding Finite Difference scheme in the over-relaxation regime

We can now switch to the analysis performed on the corresponding Finite Difference scheme. Following the usual notations for one-step explicit $(2k+1)$-points schemes [Godlewski and Raviart, 1991], that can be written under the form $\mathsf{u}_j^{n+1} = H(\mathsf{u}_{j-k}^n, \dots, \mathsf{u}_{j+k}^n)$ for $n \in \mathbb{N}$ and $j \in \mathbb{Z}$, which are said to be monotone if H is a monotone non-decreasing function of each argument, then we write (11.6) as

$$\mathsf{u}_j^{n+1} = H(\mathsf{u}_{j-1}^n, \mathsf{u}_{j+1}^n, \mathsf{u}_j^{n-1}), \qquad H(a,b,c) = \Big(1 - \frac{s}{2}(1-C)\Big)a + \Big(1 - \frac{s}{2}(1+C)\Big)b + (s-1)c. \tag{11.7}$$

To make the link with the previous discussion on $\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}$, it is not surprising that we have

$$\frac{\partial H}{\partial a}(a,b,c) = (\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{11}, \qquad \frac{\partial H}{\partial b}(a,b,c) = (\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{22}, \qquad \frac{\partial H}{\partial c}(a,b,c) = -\det(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}).$$

Proposition 11.3.2: Monotone Finite Difference scheme

In the linear setting, the Finite Difference scheme (11.6) is monotone according to the extension of the definition for the one-step explicit schemes, namely H defined by (11.7) is monotone non-decreasing with respect to each of its arguments, if and only if

$$|C| \leq 1, \qquad 1 \leq s \leq \frac{2}{1+|C|},$$

or equivalently

$$s \in [1,2], \qquad \text{and} \qquad |C| \leq \frac{2-s}{s}, \tag{FD ≥ 0}$$

where (FD ≥ 0) will be a shorthand for this "monotone Finite Difference scheme" condition.

Proof. The condition $\partial_c H(a,b,c) \geq 0$ gives $s \geq 1$. The remaining one both comes from $\partial_a H(a,b,c) = (\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{11} \geq 0$ and $\partial_b H(a,b,c) = (\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M})_{22} \geq 0$. Simple algebraic computations yield the equivalent condition. □

This result says that if we use the straightforward extension of the standard definition of monotonicity to the multi-step corresponding Finite Difference scheme, then it can be monotone only in the OR regime.

Remark 11.3.3 (On the limitations of this straightforward extension of monotonicity)**.** *In the monotonicity for multi-step scheme Proposition 11.3.2, we treat all the time steps in the past and the data shifted in space as "equal". This is the reason why it catches only the OR scheme, whereas we would like also to say something on the UR regime.*

In order to study the UR, one has to take the initialization into account, climbing up to the initial time step. Indeed, the initialization provides a certain initial structure to the numerical solution, putting it on a sort of "manifold", which ensures to have certain properties in the UR regime.

We can provide a L^∞ stability result which is agnostic of the initialization scheme, but still requires to be in the OR regime.

Proposition 11.3.3

Under the condition (FD ≥ 0), we have

$$\|\mathrm{u}^n\|_{\ell^\infty} \leq \max\big(\|\mathrm{u}^1\|_{\ell^\infty}, \|\mathrm{u}^0\|_{\ell^\infty}\big).$$

Proof. The proof is achieved by induction, using the fact that the scheme renders a convex combination of the datum at the previous two time steps. □

To see why we cannot handle the UR regime in the same straightforward way, consider a bounded domain discretized with N_x points, endowed with periodic boundary conditions. As done in Section 9.3 and [Strikwerda, 2004, Chapter 7], we can consider to write a multi-step scheme as a one step scheme by means of a block companion matrix. This boils down to considering $\mathbf{u}^n := (\mathrm{u}_1^n, \dots, \mathrm{u}_{N_x}^n, \mathrm{u}_1^{n-1}, \dots, \mathrm{u}_{N_x}^{n-1})^\mathrm{t}$ so that the bulk scheme (11.6) reads

$$\mathbf{u}^{n+1} = \boldsymbol{Q}\mathbf{u}^n, \qquad \text{where} \qquad \boldsymbol{Q} = \begin{bmatrix} \boldsymbol{Q}_0 & \boldsymbol{Q}_{-1} \\ \boldsymbol{I}_{N_x} & \mathbf{0}_{N_x} \end{bmatrix},$$

with circulant matrix

$$\boldsymbol{Q}_0 = \mathrm{circ}\begin{bmatrix} 0 \\ 1 - \frac{s}{2} + \frac{sC}{2} \\ 0 \\ \vdots \\ 0 \\ 1 - \frac{s}{2} - \frac{sC}{2} \end{bmatrix} \in \mathcal{M}_{N_x}(\mathbb{R}), \qquad \text{and} \qquad \boldsymbol{Q}_{-1} = (s-1)\boldsymbol{I}_{N_x}$$

We have two situations:

- In the OR regime, under (FD ≥ 0), the matrix $\boldsymbol{Q}$ is a stochastic matrix, with non-negative entries and sums on the rows equal to one.
- In the UR regime, the matrix $\boldsymbol{Q}$ is not stochastic.

Iterating, we have $\mathbf{u}^n = \boldsymbol{Q}^{n-1}\mathbf{u}^1$, where $\mathbf{u}^1$ contains the initial datum and the result of the first iteration, depending on the initialization scheme at hand. Stability would therefore read (7.44)

$$\|\mathbf{u}^n\|_{\ell^\infty} \leq C\|\mathbf{u}^1\|_{\ell^\infty}, \qquad n \in \mathbb{N},$$

uniformly in Δx and for any initial datum $\mathbf{u}^1$. In terms of $\boldsymbol{Q}$, this boils down to the uniform power boundedness, *cf.* Chapter 9

$$|||\boldsymbol{Q}^n|||_{\ell^\infty} \leq C, \qquad n \in \mathbb{N}, \tag{11.8}$$

uniformly in N_x. This property is difficult to check in general because the norm can initially grow in n and eventually remain bounded stabilizing to an asymptotic value. It is thus tempting to use the trivial bound $|||\boldsymbol{Q}^n|||_{\ell^\infty} \leq |||\boldsymbol{Q}|||_{\ell^\infty}^n$ and conclude when having $|||\boldsymbol{Q}|||_{\ell^\infty} \leq 1$, that is—control the "long time" behavior of the scheme by its behavior at each time step. Indeed, we have that under the conditions ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0$):

$$|||\boldsymbol{Q}|||_{\ell^\infty} = \left|\frac{\partial H}{\partial a}(a,b,c)\right| + \left|\frac{\partial H}{\partial b}(a,b,c)\right| + \left|\frac{\partial H}{\partial c}(a,b,c)\right| = \left|1 - \frac{s}{2} - \frac{sC}{2}\right| + \left|1 - \frac{s}{2} + \frac{sC}{2}\right| + |s-1|$$

$$\overset{(\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}\geq 0)}{=} 2-s+|s-1| = \begin{cases} 3-2s, & s\in[0,1], \quad (\textit{i.e. UR}), \\ 1, & s\in[1,2], \quad (\textit{i.e. OR}). \end{cases}$$

Hence we see that the trivial stability condition $\|\!|\boldsymbol{Q}|\!\|_{\ell^\infty} \leq 1$ holds only in the OR regime. Still, this does not mean—see Section 11.5.3—that the uniform power boundedness does not hold for UR, but things are not so straightforward.

We can prove the following maximum principle, which indeed coincides with the result from [Cheng, 2003, Section 6.3.3].

Proposition 11.3.4: Maximum principle and L^∞ stability

Consider the linear setting. Let $\mathsf{u}_j^0 \in [\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0]$ where $\underline{\mathsf{u}}^0 := \inf_{j\in\mathbb{Z}} \mathsf{u}_j^0$ and $\overline{\mathsf{u}}^0 := \sup_{j\in\mathbb{Z}} \mathsf{u}_j^0$ for every $j\in\mathbb{Z}$, then using the initial scheme (11.5) and under the condition (FD ≥ 0), we have

$$\mathsf{u}_j^n \in [\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0], \qquad n\in\mathbb{N}, \quad j\in\mathbb{Z}. \tag{11.9}$$

Consequently, the following L^∞ stability estimate holds

$$\|\mathsf{u}^n\|_{\ell^\infty} \leq \|\mathsf{u}^0\|_{\ell^\infty}, \qquad n\in\mathbb{N}.$$

Proof. The proof is achieved by induction, using the fact that we have a convex combination. At the initial time, where the initial scheme (11.5) is used, we observed that (FD ≥ 0) implies that $|C|\leq 1$, thus that the coefficients in (11.5) are non-negative and sum to one. □

Remark 11.3.4. *We stated Proposition 11.3.4 taking the fact that we use the initialization scheme (11.5) into account. However, since the monotonicity property is agnostic of the initial condition, we could state it without specifying the initial scheme by introducing $\underline{\mathsf{u}}^1 := \inf_{j\in\mathbb{Z}} \mathsf{u}_j^1$ and $\overline{\mathsf{u}}^1 := \sup_{j\in\mathbb{Z}} \mathsf{u}_j^1$ computed with whatever initialization scheme and saying that under (FD ≥ 0), we have*

$$\mathsf{u}_j^n \in [\min(\underline{\mathsf{u}}^0, \underline{\mathsf{u}}^1), \max(\overline{\mathsf{u}}^0, \overline{\mathsf{u}}^1)], \qquad n\in\mathbb{N}, \quad j\in\mathbb{Z}. \tag{11.10}$$

We observe that the proof of this result is straightforward and furthermore not entirely satisfying—as for the notion of monotonicity that we have utilized—because of the following reasons.

Remark 11.3.5 (Unsatisfactory notion of monotonicity).

- *In a more general non-linear framework, [Caetano et al., 2023] have proved that the existence of an invariant compact set (11.9) by the scheme holds using the initial scheme (11.5) and under the UR condition*

$$s\in[0,1], \qquad \textit{and} \qquad |C|\leq 1. \tag{LBM ≥ 0}$$

  In what follows, (LBM ≥ 0) will be a shorthand for the regime studied in [Caetano et al., 2023]. However, those authors were unable to prove that we have the same in (FD ≥ 0), thus for the OR framework. Their proof relies on convexity arguments, which are nevertheless applied to the original lattice Boltzmann scheme. We observe that their proof is probably unable to reach the area by (FD ≥ 0) because they do not take advantage of the spatial behavior of the scheme via *the stream phase (11.4) as we do. On the other hand, the limitation of our proof is that we treat all the previous times (all the arguments of H, in particular the last one) in the same way, neglecting that they correspond to different times and we do not have two preferential directions for information propagation given by the discrete velocities.*

- *[Dellacherie, 2014, Proposition 5.1] proves that (11.9) holds using the initial scheme (11.5) (or other schemes, indeed) under the conditions ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}\geq 0$), thus both in the OR and the UR regimes. Still, their proof is limited to a linear framework that we shall eventually try to overcome and does not rely neither on the corresponding Finite Difference formulation nor on the original lattice Boltzmann scheme, but on a numerical scheme which looks like (but is not) a lattice Boltzmann scheme. This scheme, called "LBM*" is the original lattice Boltzmann*

scheme where the moments are "measured" right after the collision, which is also the approach used by [Caetano et al., 2023] to show entropy inequalities.

- *The work of [Hundsdorfer et al., 2003] concerning time integration with multi-step methods points out that "... it is crucial to consider the linear multi-step method in combination with suitable starting procedures. ...More importantly, it turns out that the insistence on arbitrary starting vectors severely limits the class of methods for which monotonicity can be demonstrated." This claim seems to suggest that in UR case covered by [Caetano et al., 2023], which we are unable to analyze with the previous discussion since $\partial_c H(a,b,c) \leq 0$, needs to be studied in closer connection with the initial scheme (11.5), without trying to prove (11.10) and then enforce the particular initialization procedure to recover u^1 from the initial datum. We tried to follow the technique in [Hundsdorfer et al., 2003] in the linear case and we got access to a small zone in UR close to $s = 1$. However, the results were not satisfying. Observe that a vast literature concerning monotonicity properties for multi-step methods in time exists, see [Spijker, 2007, Gottlieb et al., 2009, Hundsdorfer et al., 2009].*

Remark 11.3.6. *It is interesting to observe that for this particular scheme at hand, the positivity of the collision matrix* ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0$) *seems to provide the whole area of parameters s and C where a maximum principle holds, being the union of* (LBM ≥ 0) *in the UR regime and* (FD ≥ 0) *for the OR one. Still, this is in general—besides the current scheme—false.*

11.4 Convergence of the non-linear lattice Boltzmann scheme in the over-relaxation regime

After these preliminary discussions, we here show the main result of Chapter 11 on the original lattice Boltzmann scheme by taking advantage of the corresponding Finite Difference scheme in the over-relaxation regime. Starting from the discrete numerical solution of the Finite Difference scheme, we shall show in the standard way that

1. We can extract a subsequence converging as the discretization parameters $\Delta t, \Delta x \to 0$, using Theorem 11.1.2.
2. The limit is a weak solution according to Definition 11.1.1.
3. The limit is the entropic weak solution according to Definition 11.1.3.

The key to prove the first point in this list is the existence of an invariant compact set for the solution which allows to provide the right estimates for the total variation of the discrete solution, which is defined as

$$\mathrm{TV}(\mathsf{u}) = \sum_{j\in\mathbb{Z}} |\mathsf{u}_{j+1} - \mathsf{u}_j|.$$

11.4.1 Monotonicity and maximum principle for the corresponding Finite Difference scheme

Let us start by discussing the maximum principle. Since we have taken $u^\circ \in L^\infty(\mathbb{R})$, we shall denote

$$\underline{\mathsf{u}}^0 := \inf_{j\in\mathbb{Z}} \mathsf{u}_j^0, \qquad \overline{\mathsf{u}}^0 := \sup_{j\in\mathbb{Z}} \mathsf{u}_j^0,$$

so that of course $\|\mathsf{u}^0\|_{\ell^\infty} = \max(|\underline{\mathsf{u}}^0|, |\overline{\mathsf{u}}^0|)$. Moreover, we introduce the Courant number for this choice of compact set $[\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0]$ by the initial datum, that is

$$C := \frac{\max_{\mathsf{u}\in[\underline{\mathsf{u}}^0,\overline{\mathsf{u}}^0]} |\varphi'(\mathsf{u})|}{\lambda}.$$

As it was the case for the linear setting with (11.7), we write the scheme as

$$\mathsf{u}_j^{n+1} = H(\mathsf{u}_{j-1}^n, \mathsf{u}_{j+1}^n, \mathsf{u}_j^{n-1}), \qquad H(a,b,c) = \Big(\Big(1-\frac{s}{2}\Big)a + \frac{s\varphi(a)}{2\lambda}\Big) + \Big(\Big(1-\frac{s}{2}\Big)b - \frac{s\varphi(b)}{2\lambda}\Big) + (s-1)c. \tag{11.11}$$

Thus we prove the same result as in the linear case, namely Proposition 11.3.2, which reads

Proposition 11.4.1: Monotone Finite Difference scheme

The non-linear Finite Difference scheme (11.6) is monotone, namely H in (11.11) is monotone non-decreasing with respect to each of its arguments when they belong to the compact set $[\underline{u}^0, \overline{u}^0]$, if and only if

$$|C| \leq 1, \qquad 1 \leq s \leq \frac{2}{1+|C|},$$

equivalently

$$s \in [1,2], \qquad \text{and} \qquad |C| \leq \frac{2-s}{s}.$$

Again, this monotonicity property holds uniquely in the OR regime. We can also prove the maximum principle:

Proposition 11.4.2: Maximum principle and L^∞ stability

Under condition (FD ≥ 0) and using the initialization (11.5), the solution of the non-linear Finite Difference scheme is such that

$$u_j^n \in [\underline{u}^0, \overline{u}^0], \qquad n \in \mathbb{N}, \qquad j \in \mathbb{Z}.$$

Consequently, the following stability estimate holds

$$\|u^n\|_{\ell^\infty} \leq \|u^0\|_{\ell^\infty}, \qquad n \in \mathbb{N}.$$

Proof. We proceed by induction.

- $n = 0$, the property trivially holds.
- $n = 1$. Since $u_j^0 \in [\underline{u}^0, \overline{u}^0]$ and $\varphi \in C^1(\mathbb{R})$, by the mean value theorem, for any $j \in \mathbb{Z}$, there exists $\hat{u}_j^0 \in]\underline{u}^0, \overline{u}^0[$ such that
$$\varphi(u_{j-1}^0) - \varphi(u_{j+1}^0) = \varphi'(\hat{u}_j^0)(u_{j-1}^0 - u_{j+1}^0).$$
Using this in (11.5) provides
$$u_j^1 = \frac{1}{2}\Big(1 + \frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)u_{j-1}^0 + \frac{1}{2}\Big(1 - \frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)u_{j+1}^0,$$
which is a convex combination since (FD ≥ 0) holds. Thus $u_j^1 \in [\underline{u}^0, \overline{u}^0]$.
- For $\tilde{n} \in \mathbb{N}^*$, assume that the claim holds true for $n \in [\![0, \tilde{n}]\!]$. Then, for any $j \in \mathbb{Z}$, there exists $\hat{u}_j^{\tilde{n}} \in]\underline{u}^0, \overline{u}^0[$ such that
$$\varphi(u_{j-1}^{\tilde{n}}) - \varphi(u_{j+1}^{\tilde{n}}) = \varphi'(\hat{u}_j^{\tilde{n}})(u_{j-1}^0 - u_{j+1}^0).$$
Therefore
$$u_j^{\tilde{n}+1} = \Big(1 - \frac{s}{2} + \frac{s\varphi'(\hat{u}_j^{\tilde{n}})}{2\lambda}\Big)u_{j-1}^{\tilde{n}} + \Big(1 - \frac{s}{2} - \frac{s\varphi'(\hat{u}_j^{\tilde{n}})}{2\lambda}\Big)u_{j+1}^{\tilde{n}} + (s-1)u_j^{\tilde{n}-1}.$$
This is a convex combination, thus $u_j^{\tilde{n}+1} \in [\underline{u}^0, \overline{u}^0]$.

□

11.4.2 Space and time total variation estimates for the corresponding Finite Difference scheme

We now provide total variation estimates in space and time, strongly relying on the previously established monotonicity and the maximum principle.

Remark 11.4.1. *[Godlewski and Raviart, 1991, Theorem 3.2] states that a monotone one-step scheme which can be put in conservative form is both total-variation-diminishing and L^∞ stable. In our case, the Finite Difference scheme*

(11.6) *cannot be put in a conservative form, since it is indeed genuinely multi-step. Finally, we will end up with the same kind of result, which is nevertheless not totally self-evident.*

Proposition 11.4.3: Spatial total variation estimate

Under condition (FD ≥ 0) and using the initialization (11.5), the solution of the non-linear Finite Difference scheme is such that

$$\mathrm{TV}(\mathrm{u}^n) \leq \mathrm{TV}(\mathrm{u}^0), \qquad n \in \mathbb{N}.$$

Proof. The proof is done by induction.

- $n = 0$ is trivial.
- $n = 1$. Starting from (11.5), we have

$$\begin{aligned}\mathrm{u}^1_{j+1} - \mathrm{u}^1_j &= \frac{1}{2}(\mathrm{u}^0_j + \mathrm{u}^0_{j+2}) + \frac{1}{2\lambda}(\varphi(\mathrm{u}^0_j) - \varphi(\mathrm{u}^0_{j+2})) - \frac{1}{2}(\mathrm{u}^0_{j-1} + \mathrm{u}^0_{j+1}) - \frac{1}{2\lambda}(\varphi(\mathrm{u}^0_{j-1}) - \varphi(\mathrm{u}^0_{j+1})),\\ &= \frac{1}{2}(\mathrm{u}^0_j - \mathrm{u}^0_{j-1}) + \frac{1}{2\lambda}(\varphi(\mathrm{u}^0_j) - \varphi(\mathrm{u}^0_{j-1})) + \frac{1}{2}(\mathrm{u}^0_{j+2} - \mathrm{u}^0_{j+1}) - \frac{1}{2\lambda}(\varphi(\mathrm{u}^0_{j+2}) - \varphi(\mathrm{u}^0_{j+1})).\end{aligned}$$

  Using the mean value theorem, for any $j \in \mathbb{Z}$, there exists $\tilde{\mathrm{u}}^0_j \in]\underline{\mathrm{u}}^0, \overline{\mathrm{u}}^0[$ such that

$$\varphi(\mathrm{u}^0_j) - \varphi(\mathrm{u}^0_{j-1}) = \varphi'(\tilde{\mathrm{u}}^0_j)(\mathrm{u}^0_j - \mathrm{u}^0_{j-1}).$$

  Thus

$$\mathrm{u}^1_{j+1} - \mathrm{u}^1_j = \frac{1}{2}\Big(1 + \frac{\varphi'(\tilde{\mathrm{u}}^0_j)}{\lambda}\Big)(\mathrm{u}^0_j - \mathrm{u}^0_{j-1}) + \frac{1}{2}\Big(1 - \frac{\varphi'(\tilde{\mathrm{u}}^0_{j+2})}{\lambda}\Big)(\mathrm{u}^0_{j+2} - \mathrm{u}^0_{j+1}).$$

  The coefficients are positive thus by triangle inequality

$$|\mathrm{u}^1_{j+1} - \mathrm{u}^1_j| \leq \frac{1}{2}\Big(1 + \frac{\varphi'(\tilde{\mathrm{u}}^0_j)}{\lambda}\Big)|\mathrm{u}^0_j - \mathrm{u}^0_{j-1}| + \frac{1}{2}\Big(1 - \frac{\varphi'(\tilde{\mathrm{u}}^0_{j+2})}{\lambda}\Big)|\mathrm{u}^0_{j+2} - \mathrm{u}^0_{j+1}|.$$

  Observe that the series associated with each term of this right hand side converge, since we have bounded total variation at time zero since we assumed that $u^\circ \in \mathrm{BV}(\mathbb{R})$. Hence we can use the associative property with series. Summing over $j \in \mathbb{Z}$ and changing indices

$$\mathrm{TV}(\mathrm{u}^1) = \sum_{j\in\mathbb{Z}} |\mathrm{u}^1_{j+1} - \mathrm{u}^1_j| \leq \mathrm{TV}(\mathrm{u}^0) + \frac{1}{2\lambda}\sum_{j\in\mathbb{Z}} \varphi'(\tilde{\mathrm{u}}^0_j)|\mathrm{u}^0_j - \mathrm{u}^0_{j-1}| - \frac{1}{2\lambda}\sum_{j\in\mathbb{Z}} \varphi'(\tilde{\mathrm{u}}^0_{j+2})|\mathrm{u}^0_{j+2} - \mathrm{u}^0_{j+1}| = \mathrm{TV}(\mathrm{u}^0).$$

- Let $\tilde{n} \in \mathbb{N}^*$ and assume that the claim holds for any $n \in [\![0, \tilde{n}]\!]$. Then

$$\begin{aligned}\mathrm{u}^{\tilde{n}+1}_{j+1} - \mathrm{u}^{\tilde{n}+1}_j =& \Big(1 - \frac{s}{2}\Big)(\mathrm{u}^{\tilde{n}}_j - \mathrm{u}^{\tilde{n}}_{j-1}) + \frac{s}{2\lambda}\big(\varphi(\mathrm{u}^{\tilde{n}}_j) - \varphi(\mathrm{u}^{\tilde{n}}_{j-1})\big)\\ &+ \Big(1 - \frac{s}{2}\Big)(\mathrm{u}^{\tilde{n}}_{j+2} - \mathrm{u}^{\tilde{n}}_{j+1}) - \frac{s}{2\lambda}\big(\varphi(\mathrm{u}^{\tilde{n}}_{j+2}) - \varphi(\mathrm{u}^{\tilde{n}}_{j+1})\big) + (s-1)(\mathrm{u}^{\tilde{n}-1}_{j+1} - \mathrm{u}^{\tilde{n}-1}_j).\end{aligned}$$

  By the mean value theorem, we have $\tilde{\mathrm{u}}^{\tilde{n}}_j \in]\underline{\mathrm{u}}^0, \overline{\mathrm{u}}^0[$ such that

$$\mathrm{u}^{\tilde{n}+1}_{j+1} - \mathrm{u}^{\tilde{n}+1}_j = \Big(1 - \frac{s}{2} + \frac{s\varphi'(\tilde{\mathrm{u}}^{\tilde{n}}_j)}{2\lambda}\Big)(\mathrm{u}^{\tilde{n}}_j - \mathrm{u}^{\tilde{n}}_{j-1}) + \Big(1 - \frac{s}{2} - \frac{s\varphi'(\tilde{\mathrm{u}}^{\tilde{n}}_{j+2})}{2\lambda}\Big)(\mathrm{u}^{\tilde{n}}_{j+2} - \mathrm{u}^{\tilde{n}}_{j+1}) + (s-1)(\mathrm{u}^{\tilde{n}-1}_{j+1} - \mathrm{u}^{\tilde{n}-1}_j).$$

  Summing, using the triangle inequality and the positivity of the coefficients

$$\mathrm{TV}(\mathrm{u}^{\tilde{n}+1}) \leq (2-s)\mathrm{TV}(\mathrm{u}^{\tilde{n}}) + (s-1)\mathrm{TV}(\mathrm{u}^{\tilde{n}-1}) \leq \mathrm{TV}(\mathrm{u}^0),$$

  using the induction assumptions and the fact that we have a convex combination.

□

We now go to the total variation in time.

Proposition 11.4.4: Time total variation estimate

Under condition (FD ≥ 0) and using the initilization (11.5), the solution of the non-linear Finite Difference scheme is such that

$$\sum_{j\in\mathbb{Z}} |u_j^{n+1} - u_j^n| \le 2TV(u^0), \qquad n \in \mathbb{N}.$$

Proof. We proceed by induction.

- $n = 0$. Using the mean value theorem thanks to the maximum principle

$$u_j^1 - u_j^0 = \frac{1}{2}(u_{j-1}^0 + u_{j+1}^0) + \frac{\varphi'(\hat{u}_j^0)}{s\lambda}(u_{j-1}^0 - u_{j+1}^0) - u_j^0 = \frac{1}{2}\Big(1 + \frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)(u_{j-1}^0 - u_j^0) + \frac{1}{2}\Big(1 - \frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)(u_{j+1}^0 - u_j^0).$$

Using the triangle inequality and the positivity of the coefficients provides

$$\begin{aligned} |u_j^1 - u_j^0| &\le \frac{1}{2}\Big(1 + \frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)|u_{j-1}^0 - u_j^0| + \frac{1}{2}\Big(1 - \frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)|u_{j+1}^0 - u_j^0|, \\ &\le \frac{1}{2}\Big(1 + \frac{|\varphi'(\hat{u}_j^0)|}{\lambda}\Big)|u_{j-1}^0 - u_j^0| + \frac{1}{2}\Big(1 + \frac{|\varphi'(\hat{u}_j^0)|}{\lambda}\Big)|u_{j+1}^0 - u_j^0| \le |u_{j-1}^0 - u_j^0| + |u_{j+1}^0 - u_j^0|. \end{aligned}$$

Summing and changing indices

$$\sum_{j\in\mathbb{Z}} |u_j^1 - u_j^0| \le 2\sum_{j\in\mathbb{Z}} |u_{j+1}^0 - u_j^0| = 2TV(u^0).$$

- $n = 1$. The estimation is slightly more involved. We have

$$\begin{aligned} u_j^2 - u_j^1 =& \Big(1 - \frac{s}{2}\Big)(u_{j-1}^1 + u_{j+1}^1) + \frac{s}{2\lambda}\big(\varphi(u_{j-1}^1) - \varphi(u_{j+1}^1)\big) \\ &+ (s-1)u_j^0 - \frac{1}{2}(u_{j-1}^0 + u_{j+1}^0) - \frac{1}{2\lambda}\big(\varphi(u_{j-1}^0) - \varphi(u_{j+1}^0)\big). \end{aligned}$$

We add and subtract the same quantities in order to deal with the fact that the initial datum is taken at equilibrium, thus for $s = 1$.

$$\begin{aligned} u_j^2 - u_j^1 =& \Big(1 - \frac{s}{2}\Big)(u_{j-1}^1 + u_{j+1}^1) + \frac{s}{2\lambda}\big(\varphi(u_{j-1}^1) - \varphi(u_{j+1}^1)\big) + (s-1)u_j^0 \\ &- \Big(1 - \frac{s}{2}\Big)(u_{j-1}^0 + u_{j+1}^0) - \frac{s}{2\lambda}\big(\varphi(u_{j-1}^0) - \varphi(u_{j+1}^0)\big) \\ &+ \frac{1}{2}(1-s)(u_{j-1}^0 + u_{j+1}^0) - \frac{(1-s)}{2\lambda}\big(\varphi(u_{j-1}^0) - \varphi(u_{j+1}^0)\big). \end{aligned}$$

We utilize the mean value theorem across time steps for the same spatial point $\check{u}_j^1 \in]\underline{u}^0, \overline{u}^0[$ and in space for the first step

$$\begin{aligned} u_j^2 - u_j^1 = \Big(1 - \frac{s}{2} + \frac{s\varphi'(\check{u}_{j-1}^1)}{2\lambda}\Big)(u_{j-1}^1 - u_{j-1}^0) + \Big(1 - \frac{s}{2} - \frac{s\varphi'(\check{u}_{j+1}^1)}{2\lambda}\Big)(u_{j+1}^1 - u_{j+1}^0) \\ + \frac{1}{2}(1-s)(u_{j-1}^0 - 2u_j^0 + u_{j+1}^0) - \frac{(1-s)\varphi'(\hat{u}_j^0)}{2\lambda}(u_{j-1}^0 - u_{j+1}^0). \end{aligned}$$

Adding and subtracting with a rearrangement

$$u_j^2 - u_j^1 = \Big(1 - \frac{s}{2} + \frac{s\varphi'(\check{u}_{j-1}^1)}{2\lambda}\Big)(u_{j-1}^1 - u_{j-1}^0) + \Big(1 - \frac{s}{2} - \frac{s\varphi'(\check{u}_{j+1}^1)}{2\lambda}\Big)(u_{j+1}^1 - u_{j+1}^0)$$

$$+\frac{1}{2}(s-1)\Big(1-\frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)(u_j^0-u_{j-1}^0)+\frac{1}{2}(s-1)\Big(1+\frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)(u_j^0-u_{j+1}^0).$$

Taking the absolute value

$$\begin{aligned}
|u_j^2-u_j^1| &= \Big(1-\frac{s}{2}+\frac{s\varphi'(\check{u}_{j-1}^1)}{2\lambda}\Big)|u_{j-1}^1-u_{j-1}^0|+\Big(1-\frac{s}{2}-\frac{s\varphi'(\check{u}_{j+1}^1)}{2\lambda}\Big)|u_{j+1}^1-u_{j+1}^0|\\
&\quad+\frac{1}{2}(s-1)\Big(1-\frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)|u_j^0-u_{j-1}^0|+\frac{1}{2}(s-1)\Big(1+\frac{\varphi'(\hat{u}_j^0)}{\lambda}\Big)|u_j^0-u_{j+1}^0|,\\
&\le \Big(1-\frac{s}{2}+\frac{s\varphi'(\check{u}_{j-1}^1)}{2\lambda}\Big)|u_{j-1}^1-u_{j-1}^0|+\Big(1-\frac{s}{2}-\frac{s\varphi'(\check{u}_{j+1}^1)}{2\lambda}\Big)|u_{j+1}^1-u_{j+1}^0|\\
&\quad+\frac{1}{2}(s-1)\Big(1+\frac{|\varphi'(\hat{u}_j^0)|}{\lambda}\Big)|u_j^0-u_{j-1}^0|+\frac{1}{2}(s-1)\Big(1+\frac{|\varphi'(\hat{u}_j^0)|}{\lambda}\Big)|u_j^0-u_{j+1}^0|,\\
&\le \Big(1-\frac{s}{2}+\frac{s\varphi'(\check{u}_{j-1}^1)}{2\lambda}\Big)|u_{j-1}^1-u_{j-1}^0|+\Big(1-\frac{s}{2}-\frac{s\varphi'(\check{u}_{j+1}^1)}{2\lambda}\Big)|u_{j+1}^1-u_{j+1}^0|\\
&\quad+(s-1)|u_j^0-u_{j-1}^0|+(s-1)|u_j^0-u_{j+1}^0|.
\end{aligned}$$

Summing and using the usual change of indices

$$\sum_{j\in\mathbb{Z}}|u_j^2-u_j^1|\le(2-s)\sum_{j\in\mathbb{Z}}|u_j^1-u_j^0|+2(s-1)\mathrm{TV}(u^0)\le 2\mathrm{TV}(u^0),$$

using the result at $n=0$.

- Let $\tilde{n}\in\mathbb{N}^*$. Assume that the claim holds for any $n\in[\![0,\tilde{n}]\!]$. We have

$$\begin{aligned}
u_j^{\tilde{n}+1}-u_j^{\tilde{n}} &= \Big(1-\frac{s}{2}\Big)(u_{j-1}^{\tilde{n}}+u_{j+1}^{\tilde{n}})+\frac{s}{2\lambda}\big(\varphi(u_{j-1}^{\tilde{n}})-\varphi(u_{j+1}^{\tilde{n}})\big)+(s-1)u_j^{\tilde{n}-1}\\
&\quad-\Big(1-\frac{s}{2}\Big)(u_{j-1}^{\tilde{n}-1}+u_{j+1}^{\tilde{n}-1})-\frac{s}{2\lambda}\big(\varphi(u_{j-1}^{\tilde{n}-1})-\varphi(u_{j+1}^{\tilde{n}-1})\big)-(s-1)u_j^{\tilde{n}-2}.
\end{aligned}$$

Using the mean value theorem across time steps

$$\begin{aligned}
u_j^{\tilde{n}+1}-u_j^{\tilde{n}} &= \Big(1-\frac{s}{2}+\frac{s\varphi'(\check{u}_{j-1}^{\tilde{n}})}{2\lambda}\Big)(u_{j-1}^{\tilde{n}}-u_{j-1}^{\tilde{n}-1})+\Big(1-\frac{s}{2}-\frac{s\varphi'(\check{u}_{j+1}^{\tilde{n}})}{2\lambda}\Big)(u_{j+1}^{\tilde{n}}-u_{j+1}^{\tilde{n}-1})\\
&\quad+(s-1)(u_j^{\tilde{n}-1}-u_j^{\tilde{n}-2}).
\end{aligned}$$

Taking the absolute value, using the triangle inequality and the positivity of the coefficients yield

$$\begin{aligned}
|u_j^{\tilde{n}+1}-u_j^{\tilde{n}}| &\le \Big(1-\frac{s}{2}+\frac{s\varphi'(\check{u}_{j-1}^{\tilde{n}})}{2\lambda}\Big)|u_{j-1}^{\tilde{n}}-u_{j-1}^{\tilde{n}-1}|+\Big(1-\frac{s}{2}-\frac{s\varphi'(\check{u}_{j+1}^{\tilde{n}})}{2\lambda}\Big)|u_{j+1}^{\tilde{n}}-u_{j+1}^{\tilde{n}-1}|\\
&\quad+(s-1)|u_j^{\tilde{n}-1}-u_j^{\tilde{n}-2}|.
\end{aligned}$$

Summing and using the change of indices

$$\sum_{j\in\mathbb{Z}}|u_j^{\tilde{n}+1}-u_j^{\tilde{n}}|\le(2-s)\sum_{j\in\mathbb{Z}}|u_j^{\tilde{n}}-u_j^{\tilde{n}-1}|+(s-1)\sum_{j\in\mathbb{Z}}|u_j^{\tilde{n}-1}-u_j^{\tilde{n}-2}|\le 2\mathrm{TV}(u^0),$$

using the induction assumption and the fact the we deal with a convex combination.

□

Observe that these estimates from Proposition 11.4.3 and Proposition 11.4.4 coincide with the ones found by [Caetano et al., 2023] in the UR regime for (LBM ≥ 0).

11.4.3 Relative compactness and extraction

We can now proceed to the extraction of a subsequence.

Theorem 11.4.1

Under condition (FD ≥ 0) and using the initialization (11.5), setting

$$\mathsf{u}_{\Delta t,\Delta x}(t,x) := \sum_{n\in\mathbb{N}}\sum_{j\in\mathbb{Z}} \mathsf{u}_j^n \mathbb{1}_{[t^n,t^{n+1}[}(t)\mathbb{1}_{[x_j,x_{j+1}[}(x), \qquad (t,x)\in\mathbb{R}_+\times\mathbb{R},$$

where u^n is the solution of the corresponding Finite Difference scheme, there exists a subsequence, also denoted $(\mathsf{u}_{\Delta t,\Delta x})_{\Delta t,\Delta x}$ for brevity, and a function $\overline{u}$ such that

$$\overline{u}\in L^\infty(\mathbb{R}_+\times\mathbb{R}_+)\cap L^1([0,T]\times\mathbb{R})\cap \mathrm{BV}([0,T]\times\mathbb{R}),$$

for all $T>0$, with

$$\mathsf{u}_{\Delta t,\Delta x}\to\overline{u}, \quad \text{in} \quad L^1_{\mathrm{loc}}(\mathbb{R}_+\times\mathbb{R}) \quad \text{as} \quad \Delta t,\Delta x\to 0.$$

Proof. Let Ω be any open bounded Lipschitz set such that $\Omega\subset[0,T]\times[-C,C]$, for some suitable $T>0$ and $C>0$. Also, consider $n_T\in\mathbb{N}$ such that $(n_T-1)\Delta t\leq T<n_T\Delta t$ and $N_x\in\mathbb{N}$ such that $(N_x-1)\Delta x\leq C<N_x\Delta x$. Now let ϕ be any $\phi\in C_c^1(\Omega)$ such that $\|\phi\|_{L^\infty(\Omega)}\leq 1$, that is $|\phi(t,x)|\leq 1$ for every $(t,x)\in\Omega$. Then we have, using the fact that the test function is compactly supported and is smooth:

$$\begin{aligned}
&\iint_\Omega \mathsf{u}_{\Delta t,\Delta x}(t,x)\mathrm{div}(\phi)(t,x)\mathrm{d}x\mathrm{d}t = \int_0^T\int_{-C}^C \mathsf{u}_{\Delta t,\Delta x}(t,x)\partial_t\phi(t,x)\mathrm{d}x\mathrm{d}t + \int_0^T\int_{-C}^C \mathsf{u}_{\Delta t,\Delta x}(t,x)\partial_x\phi(t,x)\mathrm{d}x\mathrm{d}t,\\
&= \sum_{n=0}^{n_T}\sum_{|j|\leq N_x} \mathsf{u}_j^n\int_{t^n}^{t^{n+1}}\int_{x_j}^{x_{j+1}}\partial_t\phi(t,x)\mathrm{d}x\mathrm{d}t + \sum_{n=0}^{n_T}\sum_{|j|\leq N_x}\mathsf{u}_j^n\int_{t^n}^{t^{n+1}}\int_{x_j}^{x_{j+1}}\partial_x\phi(t,x)\mathrm{d}x\mathrm{d}t,\\
&= \sum_{|j|\leq N_x}\int_{x_j}^{x_{j+1}}\sum_{n=0}^{n_T}\mathsf{u}_j^n(\phi(t^{n+1},x)-\phi(t^n,x))\mathrm{d}x + \sum_{n=0}^{n_T}\int_{t^n}^{t^{n+1}}\sum_{|j|\leq N_x}\mathsf{u}_j^n(\phi(t,x_{j+1})-\phi(t,x_j))\mathrm{d}t.
\end{aligned}$$

Then, we can use the summation by parts formula in space and time, without caring about the boundary terms since the test function is compactly supported

$$\begin{aligned}
&\iint_\Omega \mathsf{u}_{\Delta t,\Delta x}(t,x)\mathrm{div}(\phi)(t,x)\mathrm{d}x\mathrm{d}t\\
&= \sum_{|j|\leq N_x}\int_{x_j}^{x_{j+1}}\sum_{n=1}^{n_T}(\mathsf{u}_j^{n-1}-\mathsf{u}_j^n)\phi(t^n,x)\mathrm{d}x + \sum_{n=0}^{n_T}\int_{t^n}^{t^{n+1}}\sum_{j=-N_x+1}^{N_x}(\mathsf{u}_{j-1}^n-\mathsf{u}_j^n)\phi(t,x_j)\mathrm{d}t,\\
&\leq \sum_{|j|\leq N_x}\int_{x_j}^{x_{j+1}}\sum_{n=1}^{n_T}|\mathsf{u}_j^{n-1}-\mathsf{u}_j^n||\phi(t^n,x)|\mathrm{d}x + \sum_{n=0}^{n_T}\int_{t^n}^{t^{n+1}}\sum_{j=-N_x+1}^{N_x}|\mathsf{u}_{j-1}^n-\mathsf{u}_j^n||\phi(t,x_j)|\mathrm{d}t,\\
&\leq \sum_{|j|\leq N_x}\sum_{n=1}^{n_T}|\mathsf{u}_j^{n-1}-\mathsf{u}_j^n|\int_{x_j}^{x_{j+1}}\mathrm{d}x + \sum_{n=0}^{n_T}\sum_{j=-N_x+1}^{N_x}|\mathsf{u}_{j-1}^n-\mathsf{u}_j^n|\int_{t^n}^{t^{n+1}}\mathrm{d}t,\\
&\leq \Delta x\sum_{n=0}^{n_T-1}\sum_{|j|\leq N_x}|\mathsf{u}_j^{n+1}-\mathsf{u}_j^n| + \Delta t\sum_{n=0}^{n_T}\sum_{|j|\leq N_x}|\mathsf{u}_{j+1}^n-\mathsf{u}_j^n|.
\end{aligned}$$

Taking the supremum over the test functions provides

$$|\mathsf{u}_{\Delta t,\Delta x}|_{\mathrm{BV}(\Omega)}\leq \Delta x\sum_{n=0}^{n_T-1}\sum_{|j|\leq N_x}|\mathsf{u}_j^{n+1}-\mathsf{u}_j^n| + \Delta t\sum_{n=0}^{n_T}\sum_{|j|\leq N_x}|\mathsf{u}_{j+1}^n-\mathsf{u}_j^n|.$$

We therefore obtain, using Proposition 11.4.3 and Proposition 11.4.4

$$\begin{aligned}
|\mathsf{u}_{\Delta t,\Delta x}|_{\mathrm{BV}(\Omega)} &\leq \Delta x\sum_{n=0}^{n_T-1}\sum_{j\in\mathbb{Z}}|\mathsf{u}_j^{n+1}-\mathsf{u}_j^n| + \Delta t\sum_{n=0}^{n_T}\sum_{j\in\mathbb{Z}}|\mathsf{u}_{j+1}^n-\mathsf{u}_j^n| \leq 2\lambda T\mathrm{TV}(\mathsf{u}^0) + (T+\Delta t)\mathrm{TV}(\mathsf{u}^0)\\
&= (2\lambda T + T + \Delta t)\mathrm{TV}(\mathsf{u}^0),
\end{aligned}$$

which remains bounded as $\Delta t \to 0$. Furthermore, since the discrete solution satisfies the maximum principle

$$\|\mathsf{u}_{\Delta t,\Delta x}\|_{L^1(\Omega)} = \iint_\Omega |\mathsf{u}_{\Delta t,\Delta x}(t,x)|\mathrm{d}x\mathrm{d}t \le |\Omega|\|\mathsf{u}^0\|_{\ell^\infty} \le 2TC\|\mathsf{u}^0\|_{\ell^\infty} < +\infty.$$

This shows that $\mathsf{u}_{\Delta t,\Delta x}$ is bounded in the $L^1_{\mathrm{loc}}(\Omega)\cap \mathrm{BV}(\Omega)$ norm, thus by virtue of Theorem 11.1.2, we conclude. □

From now on, we consider that we have extracted a convergent subsequence according to the previous Theorem 11.4.1.

11.4.4 Convergence to a weak solution

In order to perform the weak consistency analysis—using elementary algebra and the acoustic scaling $\Delta x/\Delta t = \lambda$—we recast the Finite Difference scheme (11.6) under the following form

$$\frac{\mathsf{u}_j^n - \mathsf{u}_j^{n-1}}{\Delta t} + \frac{1}{2\Delta x}(\varphi(\mathsf{u}_{j+1}^n) - \varphi(\mathsf{u}_{j-1}^n)) = -\frac{1}{s\Delta t}(\mathsf{u}_j^{n+1} - 2\mathsf{u}_j^n + \mathsf{u}_j^{n-1}) + \frac{\lambda}{\Delta x}\Big(\frac{1}{s} - \frac{1}{2}\Big)(\mathsf{u}_{j-1}^n - 2\mathsf{u}_j^n + \mathsf{u}_{j+1}^n), \tag{11.12}$$

for $n \in \mathbb{N}^*$ and for $j \in \mathbb{Z}$. Observe that the left hand side is a (falsely) implicit discretization of the differential operators in the target equation, whereas we expect that the terms on the right hand side shall be the vanishing ones because they respectively approximate $-\Delta t/s\partial_{tt}$ and $\lambda\Delta x(1/s-1/2)\partial_{xx}$. This formulation is non-standard for explicit methods like the one we are considering. We then prove that we have convergence—upon extraction—to a weak solution.

Theorem 11.4.2

Under condition (FD ≥ 0) and using initialization (11.5), the limit $\overline{u}$ given by Theorem 11.4.1 is a weak solution of (11.1) according to Definition 11.1.1.

Proof. Let $\phi \in C_c^1(\mathbb{R}_+ \times \mathbb{R})$ and introduce $\mathsf{p}_j^n := \phi(t^n, x_j)$ so that the piecewise constant reconstruction of the test function reads

$$\phi_{\Delta t,\Delta x}(t,x) := \sum_{n\in\mathbb{N}}\sum_{j\in\mathbb{Z}} \mathsf{p}_j^n \mathbb{1}_{[t^n,t^{n+1}[}(t)\mathbb{1}_{[x_j,x_{j+1}[}(x), \qquad (t,x) \in \mathbb{R}_+ \times \mathbb{R}.$$

Following the proof of [Godlewski and Raviart, 1991, Theorem 1.1], that is, multiplying (11.12) by $\Delta t\Delta x \mathsf{p}_j^n$ and summing for $n \ge 1$ and $j \in \mathbb{Z}$, provides

$$\begin{aligned}
&\Delta x \sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\mathsf{u}_j^n - \mathsf{u}_j^{n-1})\mathsf{p}_j^n + \frac{\Delta t}{2}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\varphi(\mathsf{u}_{j+1}^n) - \varphi(\mathsf{u}_{j-1}^n))\mathsf{p}_j^n \\
&= -\frac{\Delta x}{s}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\mathsf{u}_j^{n+1} - 2\mathsf{u}_j^n + \mathsf{u}_j^{n-1})\mathsf{p}_j^n + \lambda\Delta t\Big(\frac{1}{s} - \frac{1}{2}\Big)\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\mathsf{u}_{j-1}^n - 2\mathsf{u}_j^n + \mathsf{u}_{j+1}^n)\mathsf{p}_j^n.
\end{aligned}$$

We can use summation by parts rules for finite sums and commute sums since they are finite thanks to the compact support of the test function.

- Using the summation by parts with rule with careful treatment of the boundary

$$\begin{aligned}
&\Delta x \sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\mathsf{u}_j^n - \mathsf{u}_j^{n-1})\mathsf{p}_j^n = -\Delta x \sum_{j\in\mathbb{Z}}\sum_{n\in\mathbb{N}} \mathsf{u}_j^n(\mathsf{p}_j^{n+1} - \mathsf{p}_j^n) - \Delta x\sum_{j\in\mathbb{Z}} \mathsf{u}_j^0\mathsf{p}_j^0, \\
&= -\int_0^{+\infty}\int_{\mathbb{R}} \mathsf{u}_{\Delta t,\Delta x}(t,x)\frac{\phi_{\Delta t,\Delta x}(t+\Delta t,x) - \phi_{\Delta t,\Delta x}(t,x)}{\Delta t}\mathrm{d}x\mathrm{d}t - \int_{\mathbb{R}} \mathsf{u}_{\Delta t,\Delta x}(0,x)\phi_{\Delta t,\Delta x}(0,x)\mathrm{d}x.
\end{aligned}$$

  - Since the increment of the test function in time converges pointwise and $\mathsf{u}_{\Delta t,\Delta x}$ is bounded, thus we can pass to the limit under the sign of integral

$$\int_0^{+\infty}\int_{\mathbb{R}} \mathsf{u}_{\Delta t,\Delta x}(t,x)\Big(\frac{\phi_{\Delta t,\Delta x}(t+\Delta t,x) - \phi_{\Delta t,\Delta x}(t,x)}{\Delta t}\mathrm{d}x - \partial_t\phi(t,x)\Big)\mathrm{d}x\mathrm{d}t \to 0,$$

hence, since the subsequence converges in $L^1_{\text{loc}}(\mathbb{R}_+ \times \mathbb{R})$

$$\left|\int_0^{+\infty}\int_{\mathbb{R}} (\mathrm{u}_{\Delta t,\Delta x}(t,x) - \overline{u}(t,x))\partial_t\phi(t,x)\mathrm{d}x\mathrm{d}t\right| \leq \|\partial_t\phi\|_{L^\infty(\mathbb{R}_+\times\mathbb{R})} \iint_{\mathrm{supp}(\partial_t\phi)} |\mathrm{u}_{\Delta t,\Delta x}(t,x) - \overline{u}(t,x)|\mathrm{d}x\mathrm{d}t \to 0.$$

– By the same arguments, namely the boundedness on the numerical solution and the convergence of the piecewise approximation of the test function to the test function

$$\int_{\mathbb{R}} \mathrm{u}_{\Delta t,\Delta x}(0,x)(\phi_{\Delta t,\Delta x}(0,x) - \phi(0,x))\mathrm{d}x \to 0.$$

Then, since $\mathrm{u}_{\Delta t,\Delta x}(0,x)$ converges in $L^1_{\text{loc}}(\mathbb{R})$ to u°, see proof of Proposition 20.2 in [Eymard et al., 2000]

$$\left|\int_{\mathbb{R}} (\mathrm{u}_{\Delta t,\Delta x}(0,x) - u^\circ(x))\phi(0,x)\mathrm{d}x\right| \leq \|\phi(0,\cdot)\|_{L^\infty(\mathbb{R})} \int_{\mathrm{supp}(\phi(0,\cdot))} |(\mathrm{u}_{\Delta t,\Delta x}(0,x) - u^\circ(x)|\mathrm{d}x \to 0.$$

Overall

$$\Delta x \sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\mathrm{u}_j^n - \mathrm{u}_j^{n-1})\mathrm{p}_j^n \to -\int_0^{+\infty}\int_{\mathbb{R}} \overline{u}(t,x)\partial_t\phi(t,x)\mathrm{d}x\mathrm{d}t - \int_{\mathbb{R}} u^\circ(x)\phi(0,x)\mathrm{d}x.$$

- Using the summation by parts rule twice to deal with a centered term, we have

$$\frac{\Delta t}{2}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\varphi(\mathrm{u}_{j+1}^n) - \varphi(\mathrm{u}_{j-1}^n))\mathrm{p}_j^n = -\frac{\Delta t}{2}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} \varphi(\mathrm{u}_j^n)(\mathrm{p}_{j+1}^n - \mathrm{p}_{j-1}^n) = -\int_{\Delta t}^{+\infty}\int_{\mathbb{R}} \varphi(\mathrm{u}_{\Delta t,\Delta x}(t,x))\frac{\phi_{\Delta t,\Delta x}(t,x+\Delta x) - \phi_{\Delta t,\Delta x}(t,x-\Delta x)}{2\Delta x}\mathrm{d}x\mathrm{d}t.$$

Since we have

$$\frac{\phi_{\Delta t,\Delta x}(t,x+\Delta x) - \phi_{\Delta t,\Delta x}(t,x-\Delta x)}{2\Delta x}\mathbb{1}_{[\Delta t,+\infty[}(t) \to \partial_x\phi(t,x),$$

and the flux φ on the discrete solution is bounded, we have

$$\int_0^{+\infty}\int_{\mathbb{R}} \varphi(\mathrm{u}_{\Delta t,\Delta x}(t,x))\Big(\frac{\phi_{\Delta t,\Delta x}(t,x+\Delta x) - \phi_{\Delta t,\Delta x}(t,x-\Delta x)}{2\Delta x}\mathbb{1}_{[\Delta t,+\infty[}(t) - \partial_x\phi(t,x)\Big)\mathrm{d}x\mathrm{d}t \to 0.$$

Since the subsequence convergences in $L^1_{\text{loc}}(\mathbb{R}_+ \times \mathbb{R})$ and the flux $\varphi \in C^1(\mathbb{R})$, then

$$\left|\int_0^{+\infty}\int_{\mathbb{R}} (\varphi(\mathrm{u}_{\Delta t,\Delta x}(t,x)) - \varphi(\overline{u}(t,x)))\partial_x\phi(t,x)\mathrm{d}x\mathrm{d}t\right| \leq \|\partial_x\phi\|_{L^\infty(\mathbb{R}_+\times\mathbb{R})} \iint_{\mathrm{supp}(\partial_x\phi)} |\varphi(\mathrm{u}_{\Delta t,\Delta x}(t,x)) - \varphi(\overline{u}(t,x))|\mathrm{d}x\mathrm{d}t \to 0.$$

This yields

$$\frac{\Delta t}{2}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\varphi(\mathrm{u}_{j+1}^n) - \varphi(\mathrm{u}_{j-1}^n))\mathrm{p}_j^n \to -\int_0^{+\infty}\int_{\mathbb{R}} \varphi(\overline{u}(t,x))\partial_x\phi(t,x)\mathrm{d}x\mathrm{d}t.$$

- Using the summation by parts formula

$$-\frac{\Delta x}{s}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}} (\mathrm{u}_j^{n+1} - 2\mathrm{u}_j^n + \mathrm{u}_j^{n-1})\mathrm{p}_j^n = -\frac{\Delta x}{s}\sum_{n=2}^{+\infty}\sum_{j\in\mathbb{Z}} \mathrm{u}_j^n(\mathrm{p}_j^{n+1} - 2\mathrm{p}_j^n + \mathrm{p}_j^{n-1}) - \frac{\Delta x}{s}\sum_{j\in\mathbb{Z}} \mathrm{u}_j^1(\mathrm{p}_j^2 - \mathrm{p}_j^1) + \frac{\Delta x}{s}\sum_{j\in\mathbb{Z}} (\mathrm{u}_j^1 - \mathrm{u}_j^0)\mathrm{p}_j^1.$$

We have to deal with the last term. Using the first iteration (11.5), we gain

$$\begin{aligned}\frac{\Delta x}{s}\sum_{j\in\mathbb{Z}}(\mathsf{u}_j^1-\mathsf{u}_j^0)\mathsf{p}_j^1 &= \frac{\Delta x}{2s}\sum_{j\in\mathbb{Z}}(\mathsf{u}_{j-1}^0-2\mathsf{u}_j^0+\mathsf{u}_{j+1}^0)\mathsf{p}_j^1+\frac{\Delta x}{2s\lambda}\sum_{j\in\mathbb{Z}}(\varphi(\mathsf{u}_{j-1}^0)-\varphi(\mathsf{u}_{j+1}^0))\mathsf{p}_j^1,\\ &= \frac{\Delta x}{2s}\sum_{j\in\mathbb{Z}}\mathsf{u}_j^0(\mathsf{p}_{j-1}^1-2\mathsf{p}_j^1+\mathsf{p}_{j+1}^1)+\frac{\Delta x}{2s\lambda}\sum_{j\in\mathbb{Z}}\varphi(\mathsf{u}_j^0)(\mathsf{p}_{j+1}^1-\mathsf{p}_{j-1}^1),\end{aligned}$$

using the summation by parts formula in space. Overall, we obtain

$$\begin{aligned}&-\frac{\Delta x}{s}\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}}(\mathsf{u}_j^{n+1}-2\mathsf{u}_j^n+\mathsf{u}_j^{n-1})\mathsf{p}_j^n\\ &\quad=-\frac{1}{s}\int_{2\Delta t}^{+\infty}\int_{\mathbb{R}}\mathsf{u}_{\Delta t,\Delta x}(t,x)\frac{\phi_{\Delta t,\Delta x}(t+\Delta t,x)-2\phi_{\Delta t,\Delta x}(t,x)+\phi_{\Delta t,\Delta x}(t-\Delta t,x)}{\Delta t}\mathrm{d}x\mathrm{d}t\\ &\qquad-\frac{1}{s}\int_{\mathbb{R}}\mathsf{u}_{\Delta t,\Delta x}(\Delta t,x)(\phi_{\Delta t,\Delta x}(2\Delta t,x)-\phi_{\Delta t,\Delta x}(\Delta t,x))\mathrm{d}x\\ &\quad+\frac{1}{2s}\int_{\mathbb{R}}\mathsf{u}_{\Delta t,\Delta x}(0,x)(\phi_{\Delta t,\Delta x}(\Delta t,x-\Delta x)-2\phi_{\Delta t,\Delta x}(\Delta t,x)+\phi_{\Delta t,\Delta x}(\Delta t,x+\Delta x))\mathrm{d}x\\ &\qquad+\frac{1}{2s\lambda}\int_{\mathbb{R}}\varphi(\mathsf{u}_{\Delta t,\Delta x}(0,x))(\phi_{\Delta t,\Delta x}(\Delta t,x+\Delta x)-\phi_{\Delta t,\Delta x}(\Delta t,x-\Delta x))\mathrm{d}x\to 0.\end{aligned}$$

The terms in u in this integrals are all bounded and the terms in the test function all converge to zero (either at rate Δt for the first two or Δx^2 for the last two ones), thus proceeding as at the previous points, the overall limit is zero.

- Finally

$$\begin{aligned}&\lambda\Delta t\Big(\frac{1}{s}-\frac{1}{2}\Big)\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}}(\mathsf{u}_{j-1}^n-2\mathsf{u}_j^n+\mathsf{u}_{j+1}^n)\mathsf{p}_j^n=\lambda\Delta t\Big(\frac{1}{s}-\frac{1}{2}\Big)\sum_{n\in\mathbb{N}^*}\sum_{j\in\mathbb{Z}}\mathsf{u}_j^n(\mathsf{p}_{j-1}^n-2\mathsf{p}_j^n+\mathsf{p}_{j+1}^n)\\ &=\lambda\Big(\frac{1}{s}-\frac{1}{2}\Big)\int_{\Delta t}^{+\infty}\int_{\mathbb{R}}\mathsf{u}_{\Delta t,\Delta x}(t,x)\frac{\phi_{\Delta t,\Delta x}(t,x-\Delta x)-2\phi_{\Delta t,\Delta x}(t,x)+\phi_{\Delta t,\Delta x}(t,x+\Delta x)}{\Delta x}\mathrm{d}x\mathrm{d}t\to 0,\end{aligned}$$

using the boundedness of $\mathsf{u}_{\Delta t,\Delta x}$ and the fact the expression involving the test function converges to zero at rate Δx.

□

11.4.5 Convergence to the entropic solution

Indeed, the numerical solution converges precisely to the entropic weak solution, because the numerical method fulfills a discrete entropy inequality.

Theorem 11.4.3

Under condition (FD ≥ 0) and using initialization (11.5), the limit $\overline{u}$ given by Theorem 11.4.1 is the weak entropic solution of (11.1) according to Definition 11.1.3.

Proof. Since we are in the scalar case—as previously pointed out—we can use Krushkov entropy-entropy flux pairs. Under (FD ≥ 0), the scheme function H is non-decreasing and moreover, we easily see that for any $\kappa\in\mathbb{R}$, we have $H(\kappa,\kappa,\kappa)=\kappa$, therefore the scheme is consistent. Take, for the moment, $\kappa\in[\underline{u}^0,\overline{u}^0]$ (we will eventually see what happens outside). By the monotonicity of H over $[\underline{u}^0,\overline{u}^0]$, we have that

$$\begin{aligned}\mathsf{u}_j^{n+1}=H(\mathsf{u}_{j-1}^n,\mathsf{u}_{j+1}^n,\mathsf{u}_j^{n-1})&\le H(\mathsf{u}_{j-1}^n\top\kappa,\mathsf{u}_{j+1}^n\top\kappa,\mathsf{u}_j^{n-1}\top\kappa),\\ \kappa=H(\kappa,\kappa,\kappa)&\le H(\mathsf{u}_{j-1}^n\top\kappa,\mathsf{u}_{j+1}^n\top\kappa,\mathsf{u}_j^{n-1}\top\kappa).\end{aligned}$$

Combining the inequalities and proceeding in the same way for $\perp$, we have

$$\mathsf{u}_j^{n+1}\top\kappa\le H(\mathsf{u}_{j-1}^n\top\kappa,\mathsf{u}_{j+1}^n\top\kappa,\mathsf{u}_j^{n-1}\top\kappa),$$

$$u_j^{n+1} \perp \kappa \geq H(u_{j-1}^n \perp \kappa, u_{j+1}^n \perp \kappa, u_j^{n-1} \perp \kappa).$$

Taking the difference of the inequalities, recalling that $a \top b - a \perp b = |a-b|$ and using the explicit expression of H

$$\begin{aligned}|u_j^{n+1} - \kappa| \leq \Big(1 - \frac{s}{2}\Big)(|u_{j-1}^n - \kappa| + |u_{j+1}^n - \kappa|) + (s-1)|u_j^{n-1} - \kappa| + \frac{s}{2\lambda}(\varphi(u_{j-1}^n \top \kappa) - \varphi(u_{j-1}^n \perp \kappa)) \\ - \frac{s}{2\lambda}(\varphi(u_{j+1}^n \top \kappa) - \varphi(u_{j+1}^n \perp \kappa)).\end{aligned}$$

In the case where $\kappa \in \mathbb{R} \setminus [\underline{u}^0, \overline{u}^0]$, we have two cases

- $\kappa < \underline{u}^0$. The previous inequality becomes

  $$\begin{aligned}&u_j^{n+1} - \kappa \\ &\qquad \leq \Big(1 - \frac{s}{2}\Big)(u_{j-1}^n - \kappa + u_{j+1}^n - \kappa) + (s-1)(u_j^{n-1} - \kappa) + \frac{s}{2\lambda}(\varphi(u_{j-1}^n) - \varphi(\kappa)) - \frac{s}{2\lambda}(\varphi(u_{j+1}^n) - \varphi(\kappa)),\end{aligned}$$

  thus also

  $$u_j^{n+1} \leq \Big(1 - \frac{s}{2}\Big)(u_{j-1}^n + u_{j+1}^n) + (s-1)u_j^{n-1} + \frac{s}{2\lambda}(\varphi(u_{j-1}^n - \varphi(u_{j+1}^n)),$$

  which holds with the equality.

- $\kappa > \overline{u}^0$. Here we have

  $$\begin{aligned}&-u_j^{n+1} + \kappa \\ &\qquad \leq \Big(1 - \frac{s}{2}\Big)(-u_{j-1}^n + \kappa - u_{j+1}^n + \kappa) + (s-1)(-u_j^{n-1} + \kappa) + \frac{s}{2\lambda}(\varphi(\kappa) - \varphi(u_{j-1}^n)) - \frac{s}{2\lambda}(\varphi(\kappa) - \varphi(u_{j+1}^n)),\end{aligned}$$

  hence

  $$u_j^{n+1} \geq \Big(1 - \frac{s}{2}\Big)(u_{j-1}^n + u_{j+1}^n) + (s-1)u_j^{n-1} + \frac{s}{2\lambda}(\varphi(u_{j-1}^n - \varphi(u_{j+1}^n)),$$

  which holds with the equality.

Hence, for any $\kappa \in \mathbb{R}$, we have the discrete entropy inequality of the scheme, which reads

$$\begin{aligned}|u_j^{n+1} - \kappa| \leq \Big(1 - \frac{s}{2}\Big)(|u_{j-1}^n - \kappa| + |u_{j+1}^n - \kappa|) + (s-1)|u_j^{n-1} - \kappa| + \frac{s}{2\lambda}(\varphi(u_{j-1}^n \top \kappa) - \varphi(u_{j-1}^n \perp \kappa)) \\ - \frac{s}{2\lambda}(\varphi(u_{j+1}^n \top \kappa) - \varphi(u_{j+1}^n \perp \kappa)).\end{aligned}$$

We rearrange it in a more explicit fashion, as we did for the numerical scheme, see (11.12)

$$\begin{aligned}&\frac{|u_j^n - \kappa| - |u_j^{n-1} - \kappa|}{\Delta t} + \frac{1}{2\Delta x}((\varphi(u_{j+1}^n \top \kappa) - \varphi(u_{j+1}^n \perp \kappa)) - (\varphi(u_{j-1}^n \top \kappa) - \varphi(u_{j-1}^n \perp \kappa))) \\ &\leq -\frac{1}{s\Delta t}(|u_j^{n+1} - \kappa| - 2|u_j^n - \kappa| + |u_j^{n-1} - \kappa|) + \frac{\lambda}{\Delta x}\Big(\frac{1}{s} - \frac{1}{2}\Big)(|u_{j-1}^n - \kappa| - 2|u_j^n - \kappa| + |u_{j+1}^n - \kappa|),\end{aligned}$$

thanks to the fact that $s \geq 0$. Again remark that this form of entropy inequality is non-standard. In particular, the right hand side does not have a specific sign. From this way of writing, we can repeat the proof of Theorem 11.4.2 with an inequality instead of an equality. The right hand side shall asymptotically tend to zero, rendering the weak entropy inequality on $\overline{u}$. □

11.4.6 Numerical simulations

To numerically check that we indeed converge in the L^1 norm—*cf.* Theorem 11.4.3—and that the maximum principle is fulfilled for (FD ≥ 0), we consider the Burgers flux $\varphi(u) = u^2/2$ with final time $T = 1$ and initial datum $u^\circ(x) = \mathbb{1}_{[0,1/2]}(|x|)$. We simulate using the original lattice Boltzmann scheme on the bounded domain $\Omega = [-2, 2]$

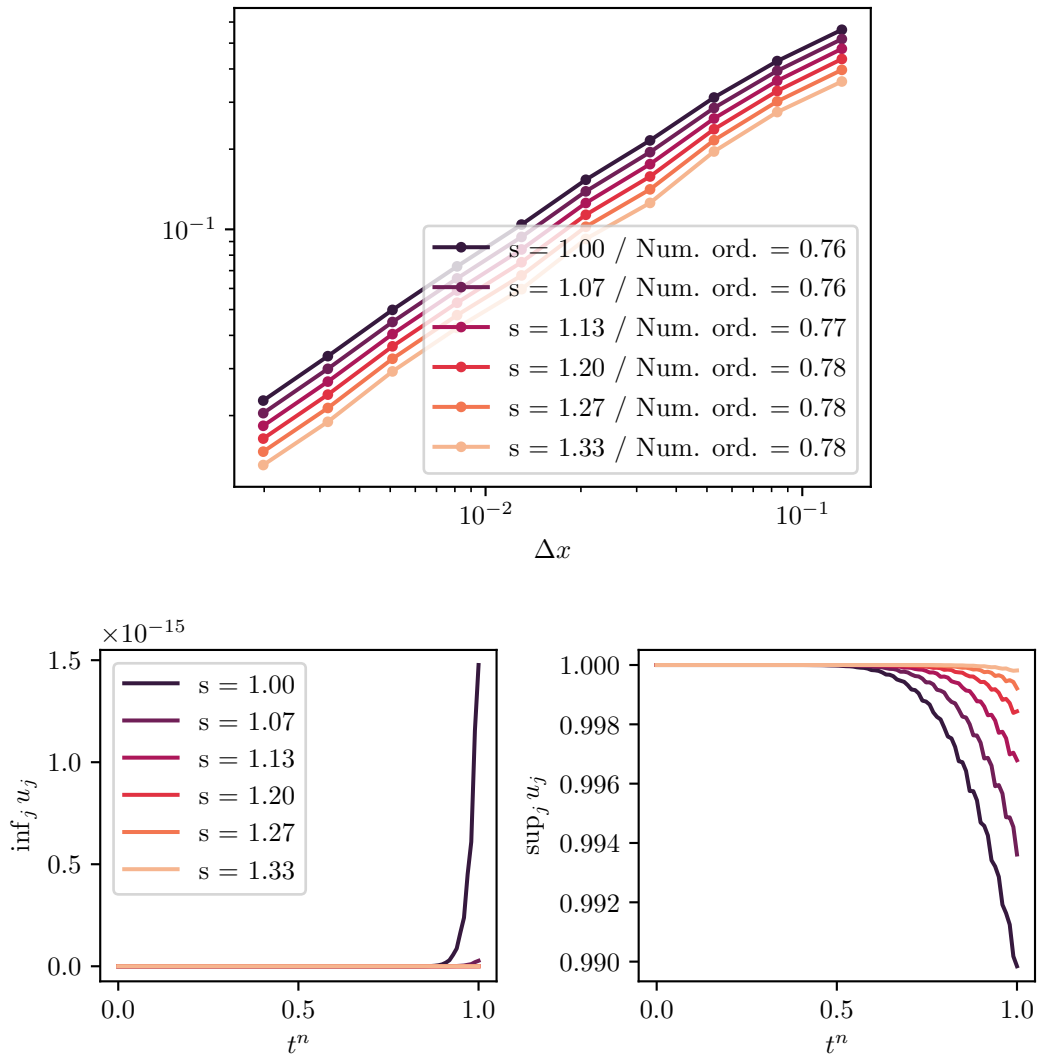


Figure 11.2: Top: error in the L^1 norm between numerical and exact solution with the numerical orders of convergence. Bottom: minimum and maximum of the numerical solution as function of the time.

endowed with periodic boundary condition and $\lambda = 2$. With this choice, the Courant number is $C = 1/2$. The error and the minimum/maximum of the solution given in Figure 11.2 confirm all the previous discussion. The convergence rate—measured numerically—is around 0.8. The more the relaxation parameter s tends to one, the more the minimum/maximum of the numerical solution shrink into the invariant compact set, due to numerical dissipation characterizing monotone schemes. Indeed, we have observed several times that numerical dissipation decreases with $s \in]0, 2]$.

11.5 Linear framework: under relaxation using the Green operators

[Caetano et al., 2023] have shown that the scheme has an invariant compact also in the UR regime. The question is: can we obtain this from the corresponding Finite Difference scheme? To answer this question, we start from a linear framework, namely we take $\varphi(u) = Vu$. In this case, the formalism of the Green functions (or Green operators) [Cheng and Lu, 1999], [Cheng, 2003, Chapter 5] or [Samarskii, 2001, Chapter 3], can be helpful to understand what is going on. In particular, it allows to "climb" back to the initial time in order to describe the influence of the initial condition. This is essential to achieve the desired properties.

11.5.1 Green operators and their properties

For the sake of notation, let us introduce the basic shift operator x, which acts in the following way: $(\mathsf{x}u)_j = u_{j-1}$ for any $j \in \mathbb{Z}$. This just corresponds to the operator x_1 that we have introduced in Chapter 7. In the sequel, let $\mathsf{d} \in \mathbb{R}[\mathsf{x}, \mathsf{x}^{-1}]$ be any Finite Difference operator in space, then we shall indicate by d_k the coefficient of the term of degree k in d. We introduce the shortcuts

$$\alpha = 1 - \frac{s}{2}(1 - C), \qquad \beta = 1 - \frac{s}{2}(1 + C), \qquad \gamma = s - 1,$$

which allow to restate the corresponding Finite Difference scheme (11.6) as

$$\mathsf{u}^{n+1} = (\alpha\mathsf{x} + \beta\overline{\mathsf{x}})\mathsf{u}^n + \gamma\mathsf{u}^{n-1}, \qquad n \in \mathbb{N}^*.$$

We introduce the first Green operator $\mathsf{G}^n \in \mathbb{R}[\mathsf{x}, \mathsf{x}^{-1}]$ at time $n \geq 2$, which corresponds to the Finite Difference operator generated by the multi-step scheme applied to the identity operator at time zero and the zero operator at time one:

$$\begin{cases} \mathsf{G}^{n+1} &= (\alpha\mathsf{x} + \beta\overline{\mathsf{x}})\mathsf{G}^n + \gamma\mathsf{G}^{n-1}, \qquad n \in \mathbb{N}^*, \\ \mathsf{G}^0 &= 1, \\ \mathsf{G}^1 &= 0. \end{cases}$$

In the same way, the second Green operator $\mathsf{K}^n \in \mathbb{R}[\mathsf{x}, \mathsf{x}^{-1}]$ at time $n \geq 2$ shall be given by

$$\begin{cases} \mathsf{K}^{n+1} &= (\alpha\mathsf{x} + \beta\overline{\mathsf{x}})\mathsf{K}^n + \gamma\mathsf{K}^{n-1}, \qquad n \in \mathbb{N}^*, \\ \mathsf{K}^0 &= 0, \\ \mathsf{K}^1 &= 1. \end{cases}$$

Example 11.5.1. *We give some example of first Green operator*

$$\begin{aligned}
\mathsf{G}^2 &= \gamma, \\
\mathsf{G}^3 &= \gamma\alpha\mathsf{x} + \gamma\beta\mathsf{x}^{-1}, \\
\mathsf{G}^4 &= \gamma\alpha^2\mathsf{x}^2 + \gamma(2\alpha\beta + \gamma) + \gamma\beta^2\mathsf{x}^{-2}, \\
\mathsf{G}^5 &= \gamma\alpha^3\mathsf{x}^3 + \gamma(3\beta\alpha^2 + 2\gamma\alpha)\mathsf{x} + \gamma(3\alpha\beta^2 + 2\gamma\beta)\mathsf{x}^{-1} + \gamma\beta^3\mathsf{x}^{-3}, \\
\mathsf{G}^6 &= \gamma\alpha^4\mathsf{x}^4 + \gamma(4\beta\alpha^3 + 3\gamma\alpha^2)\mathsf{x}^2 + \gamma(6\alpha^2\beta^2 + 6\alpha\beta\gamma + \gamma^2) + \gamma(4\alpha\beta^3 + 3\gamma\beta^2)\mathsf{x}^{-2} + \gamma\beta^4\mathsf{x}^{-4},
\end{aligned}$$

and of second Green operator

$$\begin{aligned}
\mathsf{K}^2 &= \alpha\mathsf{x} + \beta\mathsf{x}^{-1}, \\
\mathsf{K}^3 &= \alpha^2\mathsf{x}^2 + (2\alpha\beta + \gamma) + \beta^2\mathsf{x}^{-2}, \\
\mathsf{K}^4 &= \alpha^3\mathsf{x}^3 + (3\beta\alpha^2 + 2\gamma\alpha)\mathsf{x} + (3\alpha\beta^2 + 2\gamma\beta)\mathsf{x}^{-1} + \beta^3\mathsf{x}^{-3}, \\
\mathsf{K}^5 &= \alpha^4\mathsf{x}^4 + (4\beta\alpha^3 + 3\gamma\alpha^2)\mathsf{x}^2 + (6\alpha^2\beta^2 + 6\alpha\beta\gamma + \gamma^2) + (4\alpha\beta^3 + 3\gamma\beta^2)\mathsf{x}^{-2} + \beta^4\mathsf{x}^{-4}.
\end{aligned}$$

We therefore see that the coefficients G^n_k and K^n_k are tri-variate polynomials in the unknowns α, β and γ. Thus, they are also polynomial functions of s and C. The issue lies in providing a precise determination of their sign and comes from the fact that under (LBM ≥ 0), that is, in the UR regime, we have $\gamma \leq 0$. By the superposition principle, we have that the solution u^n can be expressed from the initial data u^1 and u^0 using the Green operators [Cheng and Lu, 1999]

$$\mathsf{u}^n = \mathsf{G}^n\mathsf{u}^0 + \mathsf{K}^n\mathsf{u}^1 = \mathsf{G}^n\mathsf{u}^0 + \frac{1}{\gamma}\mathsf{G}^{n+1}\mathsf{u}^1, \qquad n \geq 2,$$

or equivalently

$$u_j^n = \sum_{k=-n+2}^{n-2} G_k^n u_{j-k}^0 + \sum_{k=-n+1}^{n-1} K_k^n u_{j-k}^1 = \sum_{k=-n+2}^{n-2} G_k^n u_{j-k}^0 + \frac{1}{\gamma} \sum_{k=-n+1}^{n-1} G_k^{n+1} u_{j-k}^1, \qquad n \geq 2, \qquad j \in \mathbb{Z}.$$

We can prove that the Green operators have the following properties.

Lemma 11.5.1: Properties of the Green operators

Let $n \geq 2$, then

1. $G_k^n = 0$ if $|k| \geq n-1$, namely the coefficients are supported in $|k| \leq n-2$.
2. $G_k^n = 0$ if n and k have different parities.
3. $G_{n-2}^n = \gamma \alpha^{n-2}$ and $G_{-n+2}^n = \gamma \beta^{n-2}$.
4. The Green operators are linked by
$$K^n = \frac{1}{\gamma} G^{n+1}. \tag{11.13}$$

Proof. The proof is straightforward by induction. □

The compact support of the Green operators, whose size is proportional to n is a natural consequence of the finite speed of propagation and linked with the notion of domain of influence. The decoupling between space and time points according to the parity is a peculiarity of the D_1Q_2 scheme and of its corresponding Finite Difference scheme. Finally, the fact that there is a link (11.13) between Green operators shows that in the case (LBM ≥ 0), where $\gamma \leq 0$, the sign of the coefficients first and second Green operators are not the same.

Proposition 11.5.1: Sign of the Green operators

We have the following sign for the coefficients of the Green operators

1. Under the conditions given by (FD ≥ 0), hence in a OR setting
$$\begin{aligned} G_k^n &\geq 0, \qquad n \geq 2, \qquad k \in [\![-n+2, n-2]\!], \\ K_k^n &\geq 0, \qquad n \geq 2, \qquad k \in [\![-n+1, n-1]\!]. \end{aligned}$$
2. Under the conditions given by (LBM ≥ 0), hence in a UR setting
$$\begin{aligned} G_k^n &\leq 0, \qquad n \geq 2, \qquad k \in [\![-n+2, n-2]\!], \\ K_k^n &\geq 0, \qquad n \geq 2, \qquad k \in [\![-n+1, n-1]\!]. \end{aligned}$$

Proof. Item 1 is straightforward by induction since the coefficients α, β and γ are all non-negative and their sum is one. The initial data for both G^n and K^n are operators with positive coefficients. As far as Item 2 is concerned, we prove the inequality on K^n and we conclude for G^n using (11.13). We prove something more, namely that

$$K_k^n \geq 0, \qquad n \geq 2, \qquad k \in [\![-n+1, n-1]\!], \tag{11.14}$$
$$K_k^n \geq \alpha K_{k-1}^{n-1}, \qquad n \geq 2, \qquad k \in [\![1, n-1]\!], \tag{11.15}$$
$$K_k^n \geq \beta K_{k+1}^{n-1}, \qquad n \geq 2, \qquad k \in [\![-n+1, -1]\!]. \tag{11.16}$$

We proceed by induction. Observe that the case in which the coefficients are zero because of the same parity between time and space index is handled because we do not have strict inequalities and (11.15) and (11.16) shift along diagonals in the time-space plane, thus the space and time indices retain the same parity.

- Let $n = 2$, we have $K^2 = \alpha x + \beta x^{-1}$ and $K^1 = 1$, hence the claim is verified.

- Let $n=3$, we have $\mathsf{K}^3 = \alpha^2\mathsf{x}^2 + (2\alpha\beta+\gamma) + \beta^2\mathsf{x}^{-2}$ and $\mathsf{K}^2 = \alpha\mathsf{x}+\beta\mathsf{x}^{-1}$. We have that

$$2\alpha\beta+\gamma = \underbrace{(1-s)}_{\geq 0} + \underbrace{\frac{s^2}{2}(1-C^2)}_{\geq 0} \geq 0,$$

  thus this verifies (11.14). The remaining claims are trivially true.

- Let $\tilde{n}\geq 2$ and assume that the claims (11.14), (11.15) and (11.16) hold for any $n\in[\![2,\tilde{n}]\!]$. By definition, we have

$$\mathsf{K}_k^{\tilde{n}+1} = \alpha\mathsf{K}_{k-1}^{\tilde{n}} + \beta\mathsf{K}_{k+1}^{\tilde{n}} + \gamma\mathsf{K}_k^{\tilde{n}-1}.$$

  - $k=0$, then

$$\mathsf{K}_0^{\tilde{n}+1} = \alpha\mathsf{K}_{-1}^{\tilde{n}} + \beta\mathsf{K}_1^{\tilde{n}} + \gamma\mathsf{K}_0^{\tilde{n}-1} \geq \alpha\mathsf{K}_{-1}^{\tilde{n}} + (\alpha\beta+\gamma)\mathsf{K}_0^{\tilde{n}-1} \geq 0,$$

    where the first inequality uses that $\beta\geq 0$ and the induction assumption (11.15), whereas the second one uses the fact that $\alpha\geq 0$, $\alpha\beta+\gamma = s^2/4(1-C^2)\geq 0$ and the induction assumption (11.14).
  - $k\in[\![1,\tilde{n}]\!]$ (observe indeed that the case $k=\tilde{n}$ comes trivially from Lemma 11.5.1), then

$$\mathsf{K}_k^{\tilde{n}+1} = \alpha\mathsf{K}_{k-1}^{\tilde{n}} + \beta\mathsf{K}_{k+1}^{\tilde{n}} + \gamma\mathsf{K}_k^{\tilde{n}-1} \geq \alpha\mathsf{K}_{k-1}^{\tilde{n}} + (\alpha\beta+\gamma)\mathsf{K}_k^{\tilde{n}-1} \geq \alpha\mathsf{K}_{k-1}^{\tilde{n}} \geq 0,$$

    using the same steps than at the previous point.
  - $k\in[\![-\tilde{n},-1]\!]$ is done in the same way using (11.16).

We conclude by the strong induction principle. □

Remark 11.5.1. *Again, we see that in the case (LBM ≥ 0), the problem lies in the fact that since the coefficients of Green operators have different signs, everything works fine thanks to the specific choice of initial datum, which creates the right compensation between the signs of the two Green operators. We see that proceeding in the direction of the space-time diagonal—reminding us of the characteristic associated with each discrete velocity—allows to compensate the negative sign of γ by something positive enough. Moreover, one could prove the reversed inequalities, but what would not work is the initialization of the induction procedure, which depends on the initial iteration. This shows that the initialization plays a crucial role in this business of obtaining maximum princples for the UR regime.*

Even if of moderate interest, it is possible to explicitly compute G^n and K^n. This is finally a sort of combinatoric problem or the issue of solving a recurrence relation. Indeed, as customary for recurrence relations, we try to solve the characteristic equation, namely to look at the eigenvalues of the scheme. We have already seen (*cf.* Remark 7.5.1) that these are not Finite Difference operators. We formally write

$$\mathsf{z}^2 - (\alpha\mathsf{x}+\beta\mathsf{x}^{-1})\mathsf{z} - \gamma = 0, \qquad \text{thus} \qquad \mathsf{z} = \mathsf{z}_\pm = \frac{1}{2}\Big(\alpha\mathsf{x}+\beta\mathsf{x}^{-1} \pm \sqrt{(\alpha\mathsf{x}+\beta\mathsf{x}^{-1})^2+4\gamma}\Big),$$

where $\mathsf{z}_\pm$ do not *a priori* belong to $\mathbb{R}[\mathsf{x},\mathsf{x}^{-1}]$. This is totally formal and we do not aim at giving a precise meaning to the square root (it could be done using the discrete Fourier transform, see Section 7.7). We want to check that we have to distinct roots, namely that $(\alpha\mathsf{x}+\beta\mathsf{x}^{-1})^2+4\gamma = \alpha^2\mathsf{x}^2+2(\alpha\beta+2\gamma)+\beta^2\mathsf{x}^{-2}\neq 0$. Without loss of generality, suppose that $0\leq C\leq 1$. Assume that we are under ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}\geq 0$), then we exclude $s=1$ since it is degenerate

- $s\in[0,1[$. We have that $\beta > 1-1/2(1+C) = (1-C)\geq 0$, thus we have what we want.
- $1<s\leq 2/(1+C)$. Then $\beta = 1-s/2(1+C) > 0$ if $s<2/(1+C)$. If $s=2/(1+C)$ then $\beta=0$. In this case $\alpha = 1-s/2(1+C) = 2C/(1+C)>0$ if $C>0$. Otherwise, if $C=0$, then $\alpha=0$, but in this case $2(\alpha\beta+2\gamma) = 4\gamma = 4(s-1)>0$ since $s>1$.

Hence we have distinct formal solution except when $s=1$, which is trivial. We therefore write

$$\mathsf{G}^n = A_+(\mathsf{z}_+)^n + A_-(\mathsf{z}_-)^n,$$

where $A_\pm$ have to be determined from the initial conditision. We therefore enforce

$$\begin{cases} \mathsf{G}^0 = A_+ + A_- = 1, \\ \mathsf{G}^1 = A_+\mathsf{z}_+ + A_-\mathsf{z}_- = 0, \end{cases} \qquad \text{therefore} \qquad A_\pm = \mp\frac{\mathsf{z}_\mp}{\mathsf{z}_+ - \mathsf{z}_-}.$$

We obtain

$$\mathsf{G}^n = \frac{-\mathsf{z}_+\mathsf{z}_-\left(\mathsf{z}_+^{n-1} - \mathsf{z}_-^{n-1}\right)}{\mathsf{z}_+ - \mathsf{z}_-} = \frac{\gamma\left(\mathsf{z}_+^{n-1} - \mathsf{z}_-^{n-1}\right)}{\mathsf{z}_+ - \mathsf{z}_-},$$

using the expression of the product of the roots. We now perform a sort of kinetic change of variable, setting

$$\mathsf{z}_+ = \phi_+ + \phi_-, \qquad \mathsf{z}_- = \phi_+ - \phi_-,$$

where ϕ_+ plays the role of the positively moving population and ϕ_- that of the negatively moving population. Therefore, using the Newton binomial

$$\begin{aligned} \mathsf{G}^n &= \frac{\gamma\left((\phi_+ + \phi_-)^{n-1} - (\phi_+ - \phi_-)^{n-1}\right)}{2\phi_-} = \frac{\gamma}{2\phi_-}\left(\sum_{k=0}^{n-1}\binom{n-1}{k}\phi_+^{n-1-k}\phi_-^k - \sum_{k=0}^{n-1}\binom{n-1}{k}(-1)^k\phi_+^{n-1-k}\phi_-^k\right), \\ &= \gamma\sum_{k=0}^{n-1}\binom{n-1}{k}\phi_+^{n-1-k}\phi_-^{k-1}\frac{(1-(-1)^k)}{2} = \gamma\sum_{\substack{k=0 \\ k\,\text{odd}}}^{n-1}\binom{n-1}{k}\phi_+^{n-1-k}\phi_-^{k-1} = \gamma\sum_{k=0}^{\lfloor (n-2)/2\rfloor}\binom{n-1}{2k+1}\phi_+^{n-2-2k}\phi_-^{2k}. \end{aligned}$$

Observe that $\phi_+ = (\mathsf{z}_+ + \mathsf{z}_-)/2 = (\alpha\mathsf{x} + \beta\mathsf{x}^{-1})/2$ and that

$$\phi_-^2 = \left(\frac{\mathsf{z}_+ - \mathsf{z}_-}{2}\right)^2 = \left(\frac{\sqrt{(\alpha\mathsf{x} + \beta\mathsf{x}^{-1})^2 + 4\gamma}}{2}\right)^2 = \frac{(\alpha\mathsf{x} + \beta\mathsf{x}^{-1})^2 + 4\gamma}{4},$$

hence we obtain

$$\mathsf{G}^n = \frac{\gamma}{2^{n-2}}\sum_{k=0}^{\lfloor (n-2)/2\rfloor}\binom{n-1}{2k+1}(\alpha\mathsf{x} + \beta\mathsf{x}^{-1})^{n-2-2k}\left((\alpha\mathsf{x} + \beta\mathsf{x}^{-1})^2 + 4\gamma\right)^k. \tag{11.17}$$

Using the Newton binomial twice, first on $((\alpha\mathsf{x} + \beta\mathsf{x}^{-1})^2 + 4\gamma)^k$ and then to $(\alpha\mathsf{x} + \beta\mathsf{x}^{-1})^{n-2-2k}$ gives

$$\mathsf{G}^n = \sum_{k=0}^{\lfloor (n-2)/2\rfloor}\sum_{p=0}^{k}\sum_{q=0}^{n-2-2p}\binom{n-1}{2k+1}\binom{k}{p}\binom{n-2-2p}{q}\left(\frac{1}{2}\right)^{n-2-2p}\alpha^{n-2-2p-q}\beta^q\gamma^{p+1}\mathsf{x}^{n-2-2p-2q}.$$

Using an quite intricate change of indices and swap of sums, one gets [Cheng and Lu, 1999]

$$\mathsf{G}^n = \sum_{k=-n+2}^{n-2}\sum_{k=0}^{\lfloor (n-2-|k|)/2\rfloor}\sum_{p=k}^{\lfloor (n-2)/2\rfloor}\binom{n-1}{2p+1}\binom{p}{k}\binom{n-2-2k}{\frac{(n-2-2k-k)}{2}}\left(\frac{1}{2}\right)^{n-2-2k}\alpha^{(n-2-2k+k)/2}\beta^{(n-2-2k-k)/2}\gamma^{k+1}\mathsf{x}^k.$$

11.5.2 Total Green operators

In order to study why the particular choice of initial datum (11.5) allows to finally compensate the negative sign of γ in the UR case and yield a maximum principle, we define the Green operator corresponding to the choice of initial datum at equilibrium given by (11.5). This reads

$$\begin{cases} \mathsf{F}^{n+1} &= (\alpha\mathsf{x} + \beta\overline{\mathsf{x}})\mathsf{F}^n + \gamma\mathsf{F}^{n-1}, \qquad n \in \mathbb{N}^*, \\ \mathsf{F}^0 &= 1, \\ \mathsf{F}^1 &= \frac{1}{2}\left((1+C)\mathsf{x} + (1-C)\mathsf{x}^{-1}\right), \end{cases}$$

which also reads

$$\mathsf{F}^n = \mathsf{G}^n + \frac{1}{2}\left((1+C)\mathsf{x} + (1-C)\mathsf{x}^{-1}\right)\mathsf{K}^n = \mathsf{G}^n + \frac{1}{2\gamma}\left((1+C)\mathsf{x} + (1-C)\mathsf{x}^{-1}\right)\mathsf{G}^{n+1}, \qquad n \geq 2,$$

so we have that

$$u^n = F^n u^0, \qquad n \in \mathbb{N}.$$

We can again provide some *a priori* information on the Green operators.

Corollary 11.5.1: Properties of the total Green operator

Let $n \geq 2$, then

1. $F_k^n = 0$ if $|k| \geq n+1$, namely the coefficients are supported in $|k| \leq n$.
2. $F_k^n = 0$ if n and k have different parities.
3. $F_n^n = \frac{1}{2}(1+C)\alpha^{n-1}$ and $F_{-n}^n = \frac{1}{2}(1-C)\beta^{n-1}$.

Proof. Apply Lemma 11.5.1. □

We also have the fundamental property in terms of sign and sum to yield a maximum principle in the UR regime, which reads:

Proposition 11.5.2

Under ($\boldsymbol{M^{-1}KM \geq 0}$), thus both for (LBM ≥ 0) and (FD ≥ 0), the Green operator corresponding to the choice of initial datum at equilibrium given by (11.5) has coefficients with the following sign

$$F_k^n \geq 0, \qquad n \geq 2, \qquad k \in [\![-n, n]\!].$$

Furthermore, they all sum to one, namely

$$\sum_{k=-n}^{n} F_k^n = 1, \qquad n \in \mathbb{N}.$$

Proof. For (FD ≥ 0), the first part of the proof is straightforward since everything is positive. For (LBM ≥ 0), we have to proceed like in Proposition 11.5.1, only changing the initialization. Thus, we have to check that the claim

$$\begin{aligned} F_k^n &\geq 0, && n \geq 2, && k \in [\![-n, n]\!], \\ F_k^n &\geq \alpha F_{k-1}^{n-1}, && n \geq 2, && k \in [\![1, n]\!], \\ F_k^n &\geq \beta F_{k+1}^{n-1}, && n \geq 2, && k \in [\![-n, -1]\!]. \end{aligned}$$

hold for $n = 2, 3$. Observe that we cannot start doing the recurrence from $n = 0, 1$, because since for example $\alpha \geq \alpha|_{s=1} = (1+C)/2$, showing that $F_1^1 \geq (1+C)/2 F_0^0$, this does not allow, but quite the opposite, to conclude that also $F_1^1 \geq \alpha F_0^0$.

- $n = 2$. We have that $F^2 = \alpha(1+C)/2x^2 + (\gamma + \alpha(1-C)/2 + \beta(1+C)/2) + \beta(1-C)/2x^{-2}$.

  $$F_2^2 = \frac{1}{2}(1+C)\alpha = \alpha F_1^1 \geq 0, \qquad F_{-2}^2 = \frac{1}{2}(1-C)\beta = \beta F_{-1}^1 \geq 0.$$

  We are left to prove that

  $$F_0^2 = \gamma + \frac{1}{2}(1-C)\alpha + \frac{1}{2}(1+C)\beta \geq 0.$$

  We have

  $$F_0^2 = s\Big(1 - \frac{s}{2}\underbrace{(1+C^2)}_{\leq 2}\Big) \geq s(1-s) \geq 0.$$

- $n = 3$. The Green operator is given by

  $$F^3 = \frac{1}{2}(1+C)\alpha^2 x^3 + \Big(\frac{1}{2}(1+C)(2\alpha\beta + \gamma) + \frac{1}{2}(1-C)\alpha^2 + \gamma\alpha\Big)x$$

$$+\Big(\frac{1}{2}(1-C)(2\alpha\beta+\gamma)+\frac{1}{2}(1+C)\beta^2+\gamma\beta\Big)\mathsf{x}^{-1}+\frac{1}{2}(1-C)\beta^2\mathsf{x}^{-3}.$$

The only non-trivial parts of the proof are that $\mathsf{F}^3_{\pm1}\geq 0$ (we shall do only F^3_1 since these functions are symmetric with respect to $C=0$) and $\mathsf{F}^3_1\geq\alpha\mathsf{F}^2_0$ and $\mathsf{F}^3_{-1}\geq\beta\mathsf{F}^2_0$. For the first point, the strategy of the proof is the following.

$$\begin{cases} s\in]0,1[, \quad 0\leq C<1, \quad\to\quad \partial_s\mathsf{F}^3_1>0, \quad\to\quad \text{minimum of } \mathsf{F}^3_1 \text{ on } s=0,\\ s\in]0,1[, \quad -1<C<0, \quad\to\quad \partial_C\mathsf{F}^3_1>0, \quad\to\quad \text{minimum of } \mathsf{F}^3_1 \text{ on } C=-1, \end{cases}$$

– $0\leq C<1$. Then we have

$$\partial_s\mathsf{F}^3_1=\frac{1}{2}(1-C^2)+\frac{s}{4}(-3C^3+C^2+3C-1)=\frac{1}{2}(1-C^2)+\frac{s}{4}(1-C)(1+C)(3C-1),$$

and the issue lies in dealing with the sign of the last term.

1. If $1/3<C\leq 1$, then $\partial_s\mathsf{F}^3_1>0$, thus in this band, we are done.
2. If $0<C\leq 1/3$. By differentiating once more, we get

$$\partial_{ss}\mathsf{F}^3_1=\frac{s}{4}(1-C)(1+C)(3C-1)<0.$$

Thus the minimum of $\partial_s\mathsf{F}^3_1$ ought to be found on the boundary $s=1$. Here

$$\partial_s\mathsf{F}^3_1\geq\partial_s\mathsf{F}^3_1|_{s=1}\geq\frac{3}{18}>0.$$

Having in this area $\mathsf{F}^3_1\geq\mathsf{F}^3_1|_{s=0}=0$, we are done.

– $-1<C<0$. We have

$$\partial_C\mathsf{F}^3_1=s\Big(\frac{3}{8}s-C\Big(1-\frac{s}{4}+\frac{9sC}{8}\Big)\Big).$$

1. When $1-s/4+9sC/8>0$, that is when $C>8/(9s)(s/4-1)$, then $\partial_s\mathsf{F}^3_1>0$ and we are done.
2. When $C\leq 8/(9s)(s/4-1)$, we prove that $\partial_{CC}\mathsf{F}^3_1>0$, so the minimum of $\partial_C\mathsf{F}^3_1$ is to find on $C=-1$. We obtain

$$\partial_{CC}\mathsf{F}^3_1=-s+\frac{s^2}{4}-\frac{9s^2C}{4}\geq-s+\frac{s^2}{4}-\frac{9s^2}{4}\frac{8}{9s}\Big(\frac{s}{4}-1\Big)=s\Big(1-\frac{s}{4}\Big)>0.$$

Therefore

$$\partial_C\mathsf{F}^3_1\geq\partial_C\mathsf{F}^3_1|_{C=-1}=s(1-s)>0.$$

Having in this area $\mathsf{F}^3_1\geq\mathsf{F}^3_1|_{C=-1}=0$, we are done.

Concerning the condition $\mathsf{F}^3_1\geq\alpha\mathsf{F}^2_0$, this is verified if $(1+C)(\alpha\beta+\gamma)/2\geq 0$. Indeed

$$\frac{1}{2}(1+C)(\alpha\beta+\gamma)=\frac{1}{2}\underbrace{(1+C)}_{\geq0}\Big(\underbrace{(1-s)}_{\geq0}+\underbrace{\frac{s^2}{4}(1-C^2)}_{\geq0}\Big)\geq 0.$$

This concludes the first part of the proof. The second one can be done by induction (we have also done it using the explicit formulæ for G^n):

- For the initialization, we have $\mathsf{F}^0=1$ and $\mathsf{F}^1=(1+C)/2\mathsf{x}+(1-C)/2\mathsf{x}^{-1}$, thus the property trivially holds.
- Assume that $\sum_{k=-n}^{k=n}\mathsf{F}^n_k$ for every $n\in[\![0,\tilde{n}]\!]$. Then

$$\begin{aligned}\sum_{k=-\tilde{n}-1}^{\tilde{n}+1}\mathsf{F}^{\tilde{n}+1}_k&=\alpha\sum_{k=-\tilde{n}-1}^{\tilde{n}+1}\mathsf{F}^{\tilde{n}}_{k-1}+\beta\sum_{k=-\tilde{n}-1}^{\tilde{n}+1}\mathsf{F}^{\tilde{n}}_{k+1}+\gamma\sum_{k=-\tilde{n}-1}^{\tilde{n}+1}\mathsf{F}^{\tilde{n}-1}_k,\\ &=\alpha\sum_{k=-\tilde{n}-2}^{\tilde{n}}\mathsf{F}^{\tilde{n}}_k+\beta\sum_{k=-\tilde{n}}^{\tilde{n}+2}\mathsf{F}^{\tilde{n}}_k+\gamma\sum_{k=-\tilde{n}+1}^{\tilde{n}-1}\mathsf{F}^{\tilde{n}-1}_k,\\ &=\alpha\sum_{k=-\tilde{n}}^{\tilde{n}}\mathsf{F}^{\tilde{n}}_k+\beta\sum_{k=-\tilde{n}}^{\tilde{n}}\mathsf{F}^{\tilde{n}}_k+\gamma\sum_{k=-\tilde{n}+1}^{\tilde{n}-1}\mathsf{F}^{\tilde{n}-1}_k=\alpha+\beta+\gamma=1,\end{aligned}$$

where we have used Corollary 11.5.1 to adjust the indices.

□

11.5.3 Maximum principle and L^∞ stability

The maximum principle both for the OR and the UR in the linear framework follows automatically from the previous discussion.

Corollary 11.5.2: Maximum principle and L^∞ stability

Consider the linear setting. Let $u_j^0 \in [\underline{u}^0, \overline{u}^0]$ where $\underline{u}^0 := \inf_{j\in\mathbb{Z}} u_j^0$ and $\overline{u}^0 := \sup_{j\in\mathbb{Z}} u_j^0$ for every $j \in \mathbb{Z}$, then using the initial scheme (11.5) and under the condition ($\boldsymbol{M^{-1}KM \geq 0}$), we have

$$u_j^n \in [\underline{u}^0, \overline{u}^0], \qquad n \in \mathbb{N}, \quad j \in \mathbb{Z}. \tag{11.18}$$

Consequently, the following stability estimate holds

$$\|u^n\|_{\ell^\infty} \leq \|u^0\|_{\ell^\infty}, \qquad n \in \mathbb{N}.$$

Proof. Proposition 11.5.2 states that the solution is a convex combination of the initial datum. □

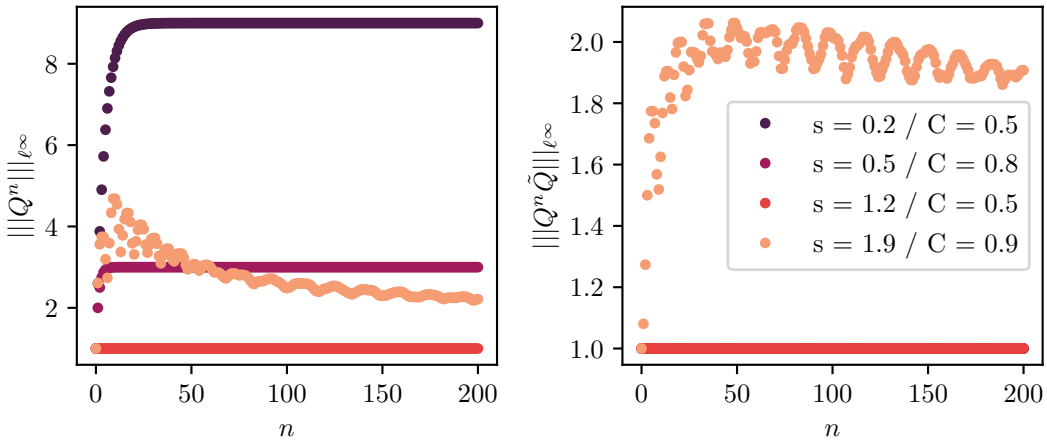


Figure 11.3: Plot of the norm $|||\boldsymbol{Q}^n|||_{\ell^\infty}$ (left) and $|||\boldsymbol{Q}^n\tilde{\boldsymbol{Q}}|||_{\ell^\infty}$ (right) as function of n for different choices of relaxation parameter and Courant number. The number of points in the lattice is $N_x = 100$.

Coming back to the discussion of Section 11.3.2 and in particular to uniform power boundedness (11.8), we can take into account the initialization (11.5) by setting $\mathbf{u}^0 = (u_1^0, \dots, u_{N_x}^0, 0, \dots, 0)^t$ and thus form the initialization step

$$\mathbf{u}^n = \boldsymbol{Q}^{n-1}\tilde{\boldsymbol{Q}}\mathbf{u}^0,$$

where $\tilde{\boldsymbol{Q}}$ takes (11.5) into account. Corollary 11.5.2 indeed shows that

$$|||\boldsymbol{Q}^n\tilde{\boldsymbol{Q}}|||_{\ell^\infty} \leq 1, \qquad \forall n \in \mathbb{N},$$

under ($\boldsymbol{M^{-1}KM \geq 0}$). This is confirmed by the numerical experiment in Figure 11.3. Here, we see that the choice of initialization (11.5) in the case of ($\boldsymbol{M^{-1}KM \geq 0}$) drastically reduces the norm (in a case, from 9 to one). We can come back to prove *a-priori* stability estimates using the Green operators, in the spirit of [Cheng and Lu, 1999,

Section 5]. Recall that

$$|u_j^n| = \left| \sum_{k=-n+2}^{n-2} G_k^n u_{j-k}^0 + \sum_{k=-n+1}^{n-1} K_k^n u_{j-k}^1 \right| \le \left(\sum_k |G_k^n| \right) \|u^0\|_{\ell^\infty} + \left(\sum_k |K_k^n| \right) \|u^1\|_{\ell^\infty}.$$

- (FD ≥ 0). We have seen that both the first and the second Green operators are positive. Using (11.17)

$$\begin{aligned}
\sum_k |G_k^n| = \sum_k G_k^n = G^n|_{x=1} &= \frac{\gamma}{2^{n-2}} \sum_{k=0}^{\lfloor (n-2)/2 \rfloor} \binom{n-1}{2k+1} (\alpha+\beta)^{n-2-2k} \big((\alpha+\beta)^2 + 4\gamma \big)^k, \\
&= \frac{(s-1)}{2^{n-2}} \sum_{k=0}^{\lfloor (n-2)/2 \rfloor} \binom{n-1}{2k+1} (2-s)^{n-2-2k} s^{2k}. \\
\sum_k |K_k^n| &= \frac{1}{2^{n-1}} \sum_{k=0}^{\lfloor (n-1)/2 \rfloor} \binom{n}{2k+1} (2-s)^{n-1-2k} s^{2k}.
\end{aligned}$$

  - **Loose estimates**. These are obtained by neglecting the parity of the terms in the previous sums.

$$\sum_k |G_k^n| = \frac{(s-1)}{2^{n-2}} \sum_{k=0}^{\lfloor (n-2)/2 \rfloor} \binom{n-1}{2k+1} (2-s)^{n-2-2k} s^{2k} \le \frac{(s-1)}{s 2^{n-2}} \sum_{k=0}^{n-1} \binom{n-1}{k} (2-s)^{n-1-k} s^k = \frac{2(s-1)}{s}.$$

$$\sum_k |K_k^n| = \frac{1}{2^{n-1}} \sum_{k=0}^{\lfloor (n-1)/2 \rfloor} \binom{n}{2k+1} (2-s)^{n-1-2k} s^{2k} \le \frac{1}{s 2^{n-1}} \sum_{k=0}^{n} \binom{n}{k} (2-s)^{n-k} s^k = \frac{2}{s}.$$

    Therefore

$$|u_j^n| \le \frac{2(s-1)}{s} \|u^0\|_{\ell^\infty} + \frac{2}{s} \|u^1\|_{\ell^\infty} \le 2 \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}),$$

    which means that we have the uniform power boundedness property $|||\boldsymbol{Q}^n|||_{\ell^\infty} \le 2$. We know that we can do better. Thus if we use initial data at equilibrium, then $|u_j^n| \le 2\|u^0\|_{\ell^\infty}$, which is $|||\boldsymbol{Q}^n \tilde{\boldsymbol{Q}}|||_{\ell^\infty} \le 2$.

  - **Sharp estimates**. These are obtained by performing the right changes of indices in the sums.

$$\ell = 2k+1, \qquad k = 0 \mapsto \ell = 1, \quad k = \left\lfloor \frac{(n-2)}{2} \right\rfloor \mapsto \ell = \begin{cases} n-1, & n \text{ even}, \\ n-2, & n \text{ odd}, \end{cases} \qquad \text{and } \ell \text{ odd}.$$

$$\begin{aligned}
\sum_k |G_k^n| = \frac{(s-1)}{2^{n-2}} \sum_{k=0}^{\lfloor (n-2)/2 \rfloor} \binom{n-1}{2k+1} (2-s)^{n-2-2k} s^{2k} &= \frac{(s-1)}{s 2^{n-1}} \sum_{\ell=0}^{n-1} \binom{n-1}{\ell} (2-s)^{n-1-\ell} s^\ell \big(1-(-1)^\ell\big), \\
&= \frac{(s-1)}{s 2^{n-1}} \sum_{\ell=0}^{n-1} \binom{n-1}{\ell} (2-s)^{n-1-\ell} s^\ell - \frac{(s-1)}{s 2^{n-1}} \sum_{\ell=0}^{n-1} \binom{n-1}{\ell} (2-s)^{n-1-\ell} (-s)^\ell, \\
&= \frac{(s-1)}{s} \Big(1 - (1-s)^{n-1} \Big).
\end{aligned}$$

    With the same way of proceeding

$$\sum_k |K_k^n| = \frac{1}{s} \Big(1 - (1-s)^n \Big).$$

    We gain

$$\begin{aligned}
|u_j^n| &\le \frac{(s-1)}{s} \overbrace{\Big(1 - (1-s)^{n-1} \Big)}^{\ge 0} \|u^0\|_{\ell^\infty} + \frac{1}{s} \overbrace{\Big(1 - (1-s)^n \Big)}^{\ge 0} \|u^1\|_{\ell^\infty}, \\
&\le \frac{(s-1)}{s} \Big(1 - (1-s)^{n-1} \Big) \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}) + \frac{1}{s} \Big(1 - (1-s)^n \Big) \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}), \\
&= \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}).
\end{aligned}$$

This yields the uniform power boundedness property $|||\boldsymbol{Q}^n|||_{\ell^\infty} \leq 1$, which proved to be optimal. Starting from the equilibrium gives $|u_j^n| \leq \|u^0\|_{\ell^\infty}$, hence $|||\boldsymbol{Q}^n\tilde{\boldsymbol{Q}}|||_{\ell^\infty} \leq 1$

- (LBM ≥ 0). We have different signs for the Green operators but the chance is that they are known for each term. Thus

$$\sum_k |G_k^n| = -\sum_k G_k^n = -G^n|_{x=1} = -\frac{(s-1)}{2^{n-2}} \sum_{k=0}^{\lfloor (n-2)/2 \rfloor} \binom{n-1}{2k+1} (2-s)^{n-2-2k} s^{2k}.$$
$$\sum_k |K_k^n| = \frac{1}{2^{n-1}} \sum_{k=0}^{\lfloor (n-1)/2 \rfloor} \binom{n}{2k+1} (2-s)^{n-1-2k} s^{2k}.$$

Following exactly the previous path, we have

  - **Loose estimates**

$$\sum_k |G_k^n| \leq -\frac{2(s-1)}{s}, \qquad \sum_k |K_k^n| \leq \frac{2}{s}.$$

Therefore

$$|u_j^n| \leq \overbrace{-\frac{2(s-1)}{s}}^{\geq 0} \|u^0\|_{\ell^\infty} + \frac{2}{s}\|u^1\|_{\ell^\infty} \leq 4\Big(\frac{1}{s} - \frac{1}{2}\Big) \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}),$$

which results in $|||\boldsymbol{Q}^n|||_{\ell^\infty} \leq 4(1/s - 1/2)$. Hence using the initial datum at equilibrium $|u_j^n| \leq 4(1/s - 1/2)\|u^0\|_{\ell^\infty}$, thus $|||\boldsymbol{Q}^n\tilde{\boldsymbol{Q}}|||_{\ell^\infty} \leq 4(1/s - 1/2)$

  - **Sharp estimates**

$$\sum_k |G_k^n| = -\frac{(s-1)}{s}\Big(1 - (1-s)^{n-1}\Big), \qquad \sum_k |K_k^n| = \frac{1}{s}\Big(1 - (1-s)^n\Big).$$

Using as it is, we gain

$$\begin{aligned} |u_j^n| &\leq -\frac{(s-1)}{s}\Big(1 - (1-s)^{n-1}\Big)\|u^0\|_{\ell^\infty} + \frac{1}{s}\Big(1 - (1-s)^n\Big)\|u^1\|_{\ell^\infty}, \\ &\leq \frac{2 - s - 2(1-s)^n}{s} \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}) \leq \frac{2-s}{s} \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}) \\ &= 2\Big(\frac{1}{s} - \frac{1}{2}\Big) \max(\|u^0\|_{\ell^\infty}, \|u^1\|_{\ell^\infty}). \end{aligned}$$

This yields $|||\boldsymbol{Q}^n|||_{\ell^\infty} \leq 2(1/s - 1/2)$. Using the initial datum at equilibrium provides $|u_j^n| \leq 2(1/s - 1/2)\|u^0\|_{\ell^\infty}$. We remark that using the previous discussion on F^n gives the sharpest possible control $|u_j^n| \leq \|u^0\|_{\ell^\infty}$, since we use the compensation between first and second Green operators. In the previous equation, this estimate cannot be obtained since, of course $(1-s)^n \leq (1-s)$, but we have the wrong sign in front of this factor. We have $2(1/s - 1/2)|_{s=0.2} = 9$ and $2(1/s - 1/2)|_{s=0.5} = 3$, thus—looking at Figure 11.3—this control appears to be optimal.

These estimates are interesting as long as one is solely interested in proving stability in the L^∞ norm, namely the uniform power boundedness of $\boldsymbol{Q}^n$. However, is order to prove that the system admits an invariant compact set, more work is needed, see Corollary 11.5.2, and the outcome highly depends on the initial condition, as previously shown. Observe that we are not able to provide stability estimates for the L^∞ norm outside the region ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0$), for the same reasons as [Cheng and Lu, 1999]. However, we cannot overcome the problem as we did for (LBM ≥ 0), since we do not know the sign of the coefficients of the Green operators but we can only use [Cheng and Lu, 1999, Equation (18) and (19)].

11.5.4 Total variation estimates

Now that—at least in the linear setting—the question of the invariant compact set has been solved using the Green functions, the new question is whether conclusions on total variation of the numerical solution can be drawn.

We start by the total variation in space. We shall assume, in analogy with the continuous case [Chambolle and Pock, 2021], that we also have the weak characterization

$$\begin{aligned}\mathrm{TV}(\mathsf{u}) &= \sup\Big\{\sum_{j\in\mathbb{Z}} \mathsf{u}_j(\mathsf{p}_j-\mathsf{p}_{j-1}) \quad : \quad (\mathsf{p}_j)_{j\in\mathbb{Z}}\subset\mathbb{R} \text{ compactly supported such that } \|\mathsf{p}\|_{\ell^\infty}\le 1\Big\},\\ &= \sup\Big\{-\sum_{j\in\mathbb{Z}} (\mathsf{u}_{j+1}-\mathsf{u}_j)\mathsf{p}_j \quad : \quad (\mathsf{p}_j)_{j\in\mathbb{Z}}\subset\mathbb{R} \text{ compactly supported such that } \|\mathsf{p}\|_{\ell^\infty}\le 1\Big\}.\end{aligned}$$

Proposition 11.5.3

Let $\mathsf{d}\in\mathbb{R}[\mathsf{x},\mathsf{x}^{-1}]$ such that $\mathsf{d}_k\ge 0$ and u such that $\mathrm{TV}(\mathsf{u})<+\infty$, then

$$\mathrm{TV}(\mathsf{du})\le\Big(\sum_{k\in\mathbb{Z}}\mathsf{d}_k\Big)\mathrm{TV}(\mathsf{u}).$$

Proof. The proof goes like it would be in the continuous case. Let $(\mathsf{p}_j)_{j\in\mathbb{Z}}\subset\mathbb{R}$ be any compactly supported sequence such that $\|\mathsf{p}\|_{\ell^\infty}\le 1$. Furthermore observe that the total variation is invariant for finite shifts, that is $\mathrm{TV}(\mathsf{x}^k\mathsf{u}) = \mathrm{TV}(\mathsf{u})$ for any $k\in\mathbb{Z}$. We have

$$\sum_{j\in\mathbb{Z}}(\mathsf{du})_j(\mathsf{p}_j-\mathsf{p}_{j-1}) = \sum_{j\in\mathbb{Z}}\sum_{k\in\mathbb{Z}}\mathsf{d}_k\mathsf{u}_{j-k}(\mathsf{p}_j-\mathsf{p}_{j-1}) = \sum_{k\in\mathbb{Z}}\overbrace{\mathsf{d}_k}^{\ge 0}\overbrace{\sum_{j\in\mathbb{Z}}\mathsf{u}_{j-k}(\mathsf{p}_j-\mathsf{p}_{j-1})}^{\le\mathrm{TV}(\mathsf{x}^k\mathsf{u})=\mathrm{TV}(\mathsf{u})}\le\Big(\sum_{k\in\mathbb{Z}}\mathsf{d}_k\Big)\mathrm{TV}(\mathsf{u}).$$

Taking the supremum over p finishes the proof. □

Corollary 11.5.3

Under ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}\ge 0$) and with initial datum at equilibrium (11.5)

$$\mathrm{TV}(\mathsf{u}^n)\le\mathrm{TV}(\mathsf{u}^0).$$

Proof. Use Proposition 11.5.2 and Proposition 11.5.3. □

As far as time is concerned, we have the following.

Proposition 11.5.4

Under ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M}\ge 0$) and with initial datum at equilibrium (11.5), we have that

$$\sum_{j\in\mathbb{Z}}|\mathsf{u}_j^{n+1}-\mathsf{u}_j^n|\le 2\|\mathsf{u}^0\|_{\ell^1}.$$

Proof. Using the Young inequality for the discrete convolution product and Proposition 11.5.2

$$\sum_{j\in\mathbb{Z}}|\mathsf{u}_j^{n+1}-\mathsf{u}_j^n| = \frac{1}{\Delta x}\|(\mathsf{F}^{n+1}-\mathsf{F}^n)\mathsf{u}^0\|_{\ell^1}\le\frac{1}{\Delta x}\|\mathsf{F}^{n+1}-\mathsf{F}^n\|_{\ell^1}\|\mathsf{u}^0\|_{\ell^1}\le\frac{1}{\Delta x}(\|\mathsf{F}^{n+1}\|_{\ell^1}+\|\mathsf{F}^n\|_{\ell^1})\|\mathsf{u}^0\|_{\ell^1} = 2\|\mathsf{u}^0\|_{\ell^1}.$$

□

It seems that this is somehow quite optimal since the coefficients of F^{n+1} are non-zero when the coefficients of F^n are zero. If the initial sequence were compactly supported, we could use a discrete analogue of the Poincaré-Wirtinger inequality with total variation [Bergounioux, 2011] to control the L^1 norm by the total variation (times a constant proportional to a length, namely the size of the support). However, we were not able to obtain an estimate like Proposition 11.4.4 where the right hand side is finite for the supposed smoothness L^∞ of the initial datum. Therefore, the time total variation estimate remains a stumbling block and should be the object of future investigations.

11.6 Maximum principle for the under-relaxation regime in the non-linear case

Now that a maximum principle and an invariant compact set have been established in the linear case for the UR regime, the question concerns the way of doing the same in the non-linear setting. The numerical scheme (11.6) and (11.5) reads as well

$$\begin{cases} \mathsf{u}^{n+1} &= sH_{\mathrm{LF}}(\mathsf{u}^n) + (1-s)H_{\mathrm{WE}}(\mathsf{u}^n, \mathsf{u}^{n-1}), \qquad n \in \mathbb{N}^*, \\ \mathsf{u}^1 &= H_{\mathrm{LF}}(\mathsf{u}^0), \end{cases}$$

where $s \in [0,1]$ and

$$H_{\mathrm{LF}}(\mathsf{u}) = \frac{1}{2}(\mathsf{x}+\mathsf{x}^{-1})\mathsf{u} + \frac{1}{2\lambda}(\mathsf{x}-\mathsf{x}^{-1})\varphi(\mathsf{u}),$$

is the Lax-Friedrichs numerical scheme with non-linear flux and

$$H_{\mathrm{WE}}(\mathsf{u},\mathsf{v}) = (\mathsf{x}+\mathsf{x}^{-1})\mathsf{u} - \mathsf{v},$$

gives a discretization of the wave equation with velocities $\pm\lambda$. It would be tempting to conclude by analyzing each scheme independently. However, the schemes are entangled and the Lax-Friedrichs scheme is non-linear, thus we cannot simply conclude by convex combination.

Let $\mathsf{a} \subset]\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0[$ be any sequence strictly contained in the invariant compact set defined by the choice of initial datum. This has to be interpreted as any sequence of potential midpoints, *cf.* the proof of Proposition 11.4.2, but for the moment, regardless of the fact that they have a precise meaning of not. We introduce the "linearized" Lax-Friedrichs scheme $H_{\mathrm{LF}}^{\ell(\mathsf{a})}$ parametrized by a, given by

$$(H_{\mathrm{LF}}^{\ell(\mathsf{a})}\mathsf{u})_j = \frac{1}{2}\Big(1 + \frac{\varphi'(\mathsf{a}_j)}{\lambda}\Big)\mathsf{u}_{j-1} + \frac{1}{2}\Big(1 - \frac{\varphi'(\mathsf{a}_j)}{\lambda}\Big)\mathsf{u}_{j+1},$$

where we use the multiplicative notation for the application of the operator to stress that it is linear in its actual argument u. We now define a parametric family of Green operators

$$\begin{cases} \mathsf{W}^{n+1} &= sH_{\mathrm{LF}}^{\ell(\mathsf{a}^n)}\mathsf{W}^n + (1-s)H_{\mathrm{WE}}(\mathsf{W}^n, \mathsf{W}^{n-1}), \qquad n \in \mathbb{N}^*, \\ \mathsf{W}^0 &= 1, \\ \mathsf{W}^1 &= H_{\mathrm{LF}}^{\ell(\mathsf{a}^0)}, \end{cases}$$

for any arbitrary $\mathsf{a}^n \subset]\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0[$ for $n \in \mathbb{N}$. Observe that we are utilizing a slight abuse of notation, since for $n \in \mathbb{N}^*$, then

$$\mathsf{W}^n = \mathsf{W}^n(\mathsf{a}^{n-1}, \dots, \mathsf{a}^0),$$

thus depend on the choice of the arbitrary linearization vectors. This again shows that we cannot consider the case $s = 0$ independently from $s = 1$, since the arguments used in $H_{\mathrm{WE}}(\cdot,\cdot)$ depend on the linearization vectors which shall be chosen also according to the Lax-Friedrichs part of the scheme.

We have previously shown by Proposition 11.5.2 that for any arbitrary $\mathsf{a}^n \subset]\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0[$ for $n \in \mathbb{N}$, we have

$$\mathsf{W}_k^n = \mathsf{W}_k^n(\mathsf{a}^{n-1}, \dots, \mathsf{a}^0) \geq 0, \qquad n \in \mathbb{N}, \qquad k \in [\![-n, n]\!],$$

and

$$\sum_{k=-n}^{n} \mathsf{W}_k^n = \sum_{k=-n}^{n} \mathsf{W}_k^n(\mathsf{a}^{n-1}, \dots, \mathsf{a}^0) = 1, \qquad n \in \mathbb{N}.$$

Now, fix $\tilde{n} \in \mathbb{N}$. Assume that for any $n \in [\![0, \tilde{n}]\!]$, the solution $\mathsf{u}^n \subset [\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0]$. Hence we have that there exist $\mathsf{b}^0, \dots, \mathsf{b}^{\tilde{n}} \subset]\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0[$ depending on $\mathsf{u}^0, \dots, \mathsf{u}^{\tilde{n}}$ (we do not stress this dependence in the following equations to keep them compact) such that

$$(\mathsf{x}-\mathsf{x}^{-1})\varphi(\mathsf{u}^n) = \varphi'(\mathsf{b}^n)(\mathsf{x}-\mathsf{x}^{-1})\mathsf{u}^n, \qquad n \in [\![0, \tilde{n}]\!],$$

and therefore

$$\mathsf{u}^{\tilde{n}+1} = \mathsf{W}^{\tilde{n}+1}(\mathsf{b}^0, \dots, \mathsf{b}^{\tilde{n}})\mathsf{u}^0,$$

which is a convex combination of the initial datum. This proves that $\mathsf{u}^{\tilde{n}+1} \subset [\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0]$. What we have essentially done is to consider an arbitrary linearization of the Lax-Friedrichs scheme and then take independently $s = 0$ and $s = 1$ and using the superposition principle. Then, we conclude taking a particular linearization sequence. We have thus proved

Proposition 11.6.1: Maximum principle and L^∞ stability

Under condition ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0$) and using the initialization (11.5), the solution of the non-linear Finite Difference scheme is such that

$$\mathsf{u}_j^n \in [\underline{\mathsf{u}}^0, \overline{\mathsf{u}}^0], \qquad n \in \mathbb{N}, \qquad j \in \mathbb{Z}.$$

Consequently, the following L^∞ stability estimate holds

$$\|\mathsf{u}^n\|_{\ell^\infty} \leq \|\mathsf{u}^0\|_{\ell^\infty}, \qquad n \in \mathbb{N}.$$

We have therefore recovered what we wanted on the whole ($\boldsymbol{M}^{-1}\boldsymbol{K}\boldsymbol{M} \geq 0$) for the non-linear case.

11.7 Conclusions and open issues

In the under-relaxation regime, we still face open issues which prevent us from concluding on the weak convergence using the same path as the over-relaxation regime.

- Total variation estimates in time. Even in the linear case, Proposition 11.5.4 is not satisfying as Proposition 11.4.4.
- Discrete entropy inequalities. We could try to proceed as in the proof of Theorem 11.4.3. We have for $s \leq 1$, for any $\kappa \in \mathbb{R}$, using the fact that the scheme function H is decreasing with respect to the last argument

$$|\mathsf{u}_j^{n+1} - \kappa| \leq \left(1 - \frac{s}{2}\right)(|\mathsf{u}_{j-1}^n - \kappa| + |\mathsf{u}_{j+1}^n - \kappa|) + (1-s)|\mathsf{u}_j^{n-1} - \kappa| + \frac{s}{2\lambda}(\varphi(\mathsf{u}_{j-1}^n \top \kappa) - \varphi(\mathsf{u}_{j-1}^n \perp \kappa)) - \frac{s}{2\lambda}(\varphi(\mathsf{u}_{j+1}^n \top \kappa) - \varphi(\mathsf{u}_{j+1}^n \perp \kappa)).$$

  Just by performing formal Taylor expansions, we see that we are not consistent with the continuous entropy inequality. The following paths and remarks could be useful to solve this issue.

  - As for finding a maximum principle, we have seen that $s \leq 1$ imposes to climb back time to reach the initial time and take the particular initialization scheme into account. It is probably the case also for the entropic features of the scheme.
  - We have to question the necessity of looking for a multi-step discrete entropy inequality. It is possible that the solution of the multi-step corresponding Finite Difference in under-relaxation and for the specific initialization at hand satisfies a one-step entropy inequality, maybe the one of the Lax-Friedrichs scheme. The rest of the scheme pertains to $H_{\mathrm{WE}}(\cdot,\cdot)$ and is linear, thus can probably be mastered using Green functions.
  - A path could be to use some comparison principle concerning dissipation, see [Tadmor, 1984]. The idea is that if one proves that a scheme is more dissipative than another one satisfying the entropy inequality (in this case, the Lax-Friedrichs scheme), the former also satisfies the entropy inequality. The problem is that here schemes are multi-step. For the standard Finite Volume schemes, it is easy to quantify the numerical viscosity without taking the time discretization into account because the diffusivity coming from time is always the same, since the flux is consistent and the time discretization has one step. In our case, the situation is probably more involved.

Once both the UR and the OR regimes are perfectly clarified for the $\mathrm{D}_1\mathrm{Q}_2$, the next step is to consider more involved schemes, such as the $\mathrm{D}_1\mathrm{Q}_3$ [Dubois et al., 2020a] and then try to formulate a general theory. Still, this cannot be done before having totally clarified the issue of the UR regime, because for every lattice Boltzmann

scheme with a symmetric set of velocities, one has

$$\det(z\boldsymbol{I} - \mathbf{A}) = \sum_{k=1}^{q} c_k z^k + \prod_{i=N+1}^{q} (1 - s_i),$$

thus we have a positive last coefficient only if $\prod_{i=N+1}^{i=q}(1 - s_i) \geq 0$. However, it seems clear that we lack of a well-established monotonicity theory—besides the trivial extension that we used—for genuinely multi-step Finite Difference schemes. Filling this hollow is a vast and stimulating subject for future research and should take the initialization routines into account.

Chapter 12

Study of boundary conditions

General context and motivation

In the entire work, we have focused very little on boundary conditions for lattice Boltzmann schemes. However, the importance of considering them is twofold. On the one hand, they arise as numerical boundary conditions, regardless of the fact that the problem we want to approximate features a boundary or not, because one cannot perform numerical computations on infinite domains. On the other hand, one may be interested in enforcing a precise physical behavior of the solution at the boundary (*i.e.* inflow, solid walls, *etc.*). This aspect has essentially to do with consistency. The issue is complicated by the fact that, since the spatial phase of the lattice Boltzmann scheme is shaped by the discrete velocities, any boundary of the domain—even those where there is no physical boundary condition to enforce—needs numerical boundary conditions. As always, the numerical scheme for a well-posed problem is "richer" than the equations it approximates and—though being consistent—can lead to the formation of numerical instabilities. It is therefore crucial to be able to select ways of enforcing boundary conditions whilst keeping the discretization stable.

State of the art

The body of literature concerning boundary conditions for lattice Boltzmann schemes is extremely vast, see [Krüger et al., 2017, Chapter 5] and references therein. Many authors have proposed boundary conditions for specific problems at hand. However, we believe that—due to the huge variety of approaches and the lack of theoretical tools—we are far from having a comprehensive and organized theoretical framework to treat this kind of issue. In the vast number of available contributions, we believe that [Dubois et al., 2015] and [Dubois et al., 2020b] are those closer to the spirit of what we are going to develop. Here, the authors propose an asymptotic expansion to study the consistency of bounce-back [Ginzbourg and Adler, 1994, Bouzidi et al., 2001] and anti-bounce-back [Ginzburg, 2005, Ginzburg et al., 2008a] conditions for the D_2Q_9 scheme. The issue of boundary conditions is an active research topic, *e.g.* the very recent enhanced conditions [Marson et al., 2021, Ginzburg et al., 2023] based on the bounce-back rule, used to impose Dirichlet conditions on the velocity field on complex geometries with second-order accuracy, preserving the locality of the scheme. Furthermore, the previous contributions point out that two-relaxation-times schemes with links—which we have investigated in Section 7.6.3 and Section 10.4.3—are of particular interest as far as enforcing boundary conditions is concerned.

Aims and structure of Chapter 12

However, due to the fact that works on boundary conditions respond to needs germane to real-world applications and are thus mainly "application driven", we lack basic but systematic studies of boundary conditions for simple lattice Boltzmann schemes. Thus, the aim of Chapter 12 is to investigate boundary conditions for two simple one-dimensional schemes as far as consistency and stability are concerned. Chapter 12 is structured as follows.

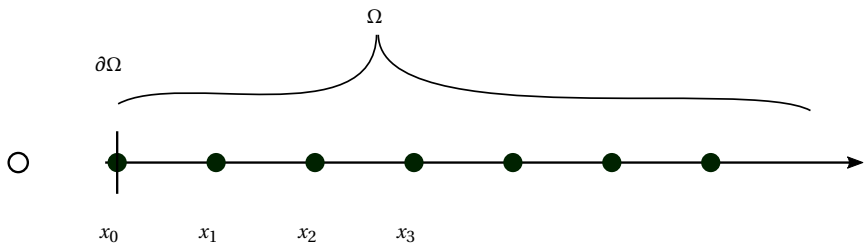


Figure 12.1: Example of discretization of a half-line problem when studying boundary conditions. The empy dot corresponds to the first point which is not part of the computational mesh.

In Section 12.1, we introduce the spatial setting we work with, which features a semi-infinite domain. Then, Section 12.2 is devoted to the study of the boundary conditions for the D_1Q_2 scheme. In order to do this, we eliminate the non-conserved moment and study the consistency of the scheme at the boundary. Furthermore, this allows us to discuss stability using the so-called GKS (Gustafsson, Kreiss, Sundström) theory. Then, we study consistency again with a formal Maxwell iteration procedure and compare the results with the previously obtained ones. The last Section 12.3 is dedicated to the study of the boundary conditions for the D_1Q_3 scheme for two conservation laws. We particularly focus on the possibility of devising transparent boundary conditions which allow to simulate a wave equation as if we were on an infinite domain, avoiding wave reflections. To this end, we eliminate the non-conserved moment and we propose a consistency analysis. The conclusions and perspectives for this small and embryonic piece of work on boundary conditions are given in Section 12.4.

Contents

12.1 Spatial setting

For this is a preliminary study, we focus on the one-dimensional case, hence we take $d = 1$. Moreover, as common when studying boundary conditions [Gustafsson et al., 1972, Coulombel, 2009, Coulombel, 2011b], we consider the half-line problem on $\Omega = \mathbb{R}_+$. The discrete points of the lattice shall be $x_j = j\Delta x$ for $j \in \mathbb{N}$, where the last point inside the discrete lattice shall be indexed by $j = 0$, *cf.* Figure 12.1. We take $x_0 = 0$, hence lying on $\partial\Omega$ so that when refining with $\Delta x \to 0$, no new point is formed between this point and $\partial\Omega$. However, the literature features cases where $\partial\Omega$ is placed half-way between lattice points or other configurations [Bouzidi et al., 2001]. We are not going to investigate this point in our contribution.

12.2 D_1Q_2 scheme with one conserved moment

We start by studying what happens on the most simple scheme, namely the D_1Q_2 scheme, *cf.* Section 1.5.1. We thus adopt the same notations as Chapter 11. As the stream phase is the only part of the algorithm which involves information coming from the neighboring cells, when $j = 0$, something must be done by imposing a "numerical" boundary condition. Besides its (numerical) role of replacing lacking pieces of information, this procedure also aims at enforcing the physics (inflow, *etc.*) at the boundary. From (11.4), we have

$$\mathsf{f}_0^{+,n+1} = \mathsf{f}_{-1}^{+,n,\star}, \qquad \mathsf{f}_0^{-,n+1} = \mathsf{f}_1^{-,n,\star}, \tag{12.1}$$

but $\mathsf{f}_{-1}^{+,n,\star}$ is not defined, since $x_{-1} = -\Delta x$ does not belong to the grid.

12.2.1 Local boundary conditions

Table 12.1: Four essential choices of local boundary conditions.

β^+	β^-	Name
1	0	0-th order extrapolation
-1	0	0-th order anti extrapolation
0	1	bounce-back [Dubois et al., 2015]
0	-1	anti-bounce-back [Dubois et al., 2020b]

The first way of replacing lacking information $\mathsf{f}_{-1}^{+,n,\star}$ that we propose is to use data at the point x_0. This reads

$$\mathsf{f}_0^{+,n+1} = \overbrace{\beta^+ \mathsf{f}_0^{+,n,\star} + \beta^- \mathsf{f}_0^{-,n,\star}}^{\text{replacing } \mathsf{f}_{-1}^{+,n,\star}} + \mathsf{S}_0^{n+1}, \tag{12.2}$$

$$\mathsf{f}_0^{-,n+1} = \mathsf{f}_1^{-,n,\star}, \tag{12.3}$$

where β^+ and β^- are the weights for the positively and negatively moving distributions in the boundary condition and S_0^{n+1} is a source term that one may wish to include in the boundary condition to make it non-homogeneous and time-dependent. Four essential choices for β^+ and β^- that we shall analyze are given in Table 12.1.

12.2.1.1 Elimination of the non-conserved moment

In order to analyze the consistency and stability of the boundary conditions (12.2) according to the choice of β^+ and β^-, the idea is to proceed like in Chapter 7: eliminate the non-conserved moment v from the formulations in order to obtain a corresponding Finite Difference scheme only on the conserved moment u. In Chapter 7, this has been done inside the domain only.

Proposition 12.2.1: Corresponding Finite Difference scheme

Under the condition

$$\beta^+ - \beta^- = \pm 1, \qquad \text{hence} \qquad \beta^- = \beta^+ \mp 1, \tag{12.4}$$

the corresponding Finite Difference scheme for the D_1Q_2 scheme (11.3) and (11.4) endowed with the boundary condition (12.2) is

$$\mathsf{u}_j^{n+1} = \frac{1}{2}(2-s)(\mathsf{u}_{j-1}^n + \mathsf{u}_{j+1}^n) - (1-s)\mathsf{u}_j^{n-1} + \frac{s}{2\lambda}(\mathsf{v}_{j-1}^{\mathrm{eq},n} - \mathsf{v}_{j+1}^{\mathrm{eq},n}), \qquad j \geq 1$$

$$\mathsf{u}_0^{n+1} = \frac{1}{2}(2\beta^+ \mp (2-s))\mathsf{u}_0^n + \frac{1}{2}(2-s)\mathsf{u}_1^n + (1-s)(\pm\beta^+ - 1)\mathsf{u}_0^{n-1} + \frac{s}{2\lambda}\left(\pm\mathsf{v}_0^{\mathrm{eq},n} - \mathsf{v}_1^{\mathrm{eq},n}\right) + \mathsf{S}_0^{n+1} \pm (1-s)\mathsf{S}_0^n. \tag{12.5}$$

Remark 12.2.1 (Where is the non-linearity?). *The origin of the constraint* (12.4) *comes from the fact of being able to recast everything only on the conserved moment* u. *The fact that this is conditionally possible seems to suggest that there is no general result such as the Cayley-Hamilton Theorem* 7.3.1 *allowing to do this in full generality. To our understanding, this comes by the fact that the boundary breaks the spatial invariance of the scheme, and one point on the lattice is not "sure" whether its neighbor is going to process the solution through the same numerical scheme. This can be interpreted as a sort of non-linearity of the lattice Boltzmann scheme introduced by the presence of a boundary.*

Proof of Proposition 12.2.1. For $j=0$, considering the collision model, we obtain

$$\mathsf{f}_0^{+,n+1} = \frac{(\beta^+ + \beta^-)}{2}\mathsf{u}_0^n + \frac{(\beta^+ - \beta^-)(1-s)}{2\lambda}\mathsf{v}_0^n + \frac{(\beta^+ - \beta^-)s}{2\lambda}\mathsf{v}_0^{\mathrm{eq},n} + \mathsf{S}_0^{n+1}, \tag{12.6}$$

$$\mathsf{f}_0^{-,n+1} = \frac{1}{2}\mathsf{u}_1^n - \frac{(1-s)}{2\lambda}\mathsf{v}_1^n - \frac{s}{2\lambda}\mathsf{v}_1^{\mathrm{eq},n}. \tag{12.7}$$

Recasting wholly on the moments using the matrix $\boldsymbol{M}$ provides

$$\mathsf{u}_0^{n+1} = \frac{1}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^n + \mathsf{u}_1^n\Big) + \frac{(1-s)}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^n - \mathsf{v}_1^n\Big) + \frac{s}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n} - \mathsf{v}_1^{\mathrm{eq},n}\Big) + \mathsf{S}_0^{n+1}, \tag{12.8}$$

$$\mathsf{v}_0^{n+1} = \frac{\lambda}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^n - \mathsf{u}_1^n\Big) + \frac{(1-s)}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_0^n + \mathsf{v}_1^n\Big) + \frac{s}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n} + \mathsf{v}_1^{\mathrm{eq},n}\Big) + \lambda\mathsf{S}_0^{n+1}. \tag{12.9}$$

The scheme inside the domain, that is, for $j \geq 1$, reads

$$\mathsf{u}_j^{n+1} = \frac{1}{2}(\mathsf{u}_{j-1}^n + \mathsf{u}_{j+1}^n) + \frac{(1-s)}{2\lambda}(\mathsf{v}_{j-1}^n - \mathsf{v}_{j+1}^n) + \frac{s}{2\lambda}(\mathsf{v}_{j-1}^{\mathrm{eq},n} - \mathsf{v}_{j+1}^{\mathrm{eq},n}), \tag{12.10}$$

$$\mathsf{v}_j^{n+1} = \frac{\lambda}{2}(\mathsf{u}_{j-1}^n - \mathsf{u}_{j+1}^n) + \frac{(1-s)}{2}(\mathsf{v}_{j-1}^n + \mathsf{v}_{j+1}^n) + \frac{s}{2}(\mathsf{v}_{j-1}^{\mathrm{eq},n} + \mathsf{v}_{j+1}^{\mathrm{eq},n}). \tag{12.11}$$

Taking (12.9) at the previous time step and at $j=1$ provides

$$\mathsf{v}_1^n = \frac{\lambda}{2}(\mathsf{u}_0^{n-1} - \mathsf{u}_2^{n-1}) + \frac{(1-s)}{2}(\mathsf{v}_0^{n-1} + \mathsf{v}_2^{n-1}) + \frac{s}{2}(\mathsf{v}_0^{\mathrm{eq},n-1} + \mathsf{v}_2^{\mathrm{eq},n-1}). \tag{12.12}$$

Writing (12.9) at the previous time step gives

$$\mathsf{v}_0^n = \frac{\lambda}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^{n-1} - \mathsf{u}_1^{n-1}\Big) + \frac{(1-s)}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{n-1} + \mathsf{v}_1^{n-1}\Big) + \frac{s}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n-1} + \mathsf{v}_1^{\mathrm{eq},n-1}\Big) + \lambda\mathsf{S}_0^n. \tag{12.13}$$

Considering $(\beta^+ - \beta^-)$(12.13) $-$ (12.12) yields

$$\begin{aligned}(\beta^+ - \beta^-)\mathsf{v}_0^n - \mathsf{v}_1^n =& \frac{\lambda}{2}\Big(((\beta^+)^2 - (\beta^-)^2 - 1)\mathsf{u}_0^{n-1} - (\beta^+ - \beta^-)\mathsf{u}_1^{n-1} + \mathsf{u}_2^{n-1}\Big)\\ &+ \frac{(1-s)}{2}\Big((\beta^+ - \beta^-)^2 - 1)\mathsf{v}_0^{n-1} + (\beta^+ - \beta^-)\mathsf{v}_1^{n-1} - \mathsf{v}_2^{n-1}\Big)\\ &+ \frac{s}{2}\Big((\beta^+ - \beta^-)^2 - 1)\mathsf{v}_0^{\mathrm{eq},n-1} + (\beta^+ - \beta^-)\mathsf{v}_1^{\mathrm{eq},n-1} - \mathsf{v}_2^{\mathrm{eq},n-1}\Big) + \lambda(\beta^+ - \beta^-)\mathsf{S}_0^n.\end{aligned} \tag{12.14}$$

The term which makes the elimination of the conserved moment difficult is $(\beta^+ - \beta^-)^2 - 1$. We observe that for the four conditions in Table 12.1, it is equal to zero. Hence, we assume that

$$(\beta^+ - \beta^-)^2 = 1, \qquad \text{hence} \qquad |\beta^+ - \beta^-| = 1.$$

Under this assumption, (12.14) becomes

$$\begin{aligned}(\beta^+ - \beta^-)\mathsf{v}_0^n - \mathsf{v}_1^n =& \frac{\lambda}{2}\Big(((\beta^+)^2 - (\beta^-)^2 - 1)\mathsf{u}_0^{n-1} - (\beta^+ - \beta^-)\mathsf{u}_1^{n-1} + \mathsf{u}_2^{n-1}\Big)\\ &+ \frac{(1-s)}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_1^{n-1} - \mathsf{v}_2^{n-1}\Big) + \frac{s}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_1^{\mathrm{eq},n-1} - \mathsf{v}_2^{\mathrm{eq},n-1}\Big) + \lambda(\beta^+ - \beta^-)\mathsf{S}_0^n.\end{aligned} \tag{12.15}$$

We write (12.8) at the previous time step and (12.10) at the previous time step for $j=1$, giving

$$\mathsf{u}_0^n = \frac{1}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^{n-1} + \mathsf{u}_1^{n-1}\Big) + \frac{(1-s)}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{n-1} - \mathsf{v}_1^{n-1}\Big) + \frac{s}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n-1} - \mathsf{v}_1^{\mathrm{eq},n-1}\Big) + \mathsf{S}_0^n, \tag{12.16}$$

and

$$\mathsf{u}_1^n = \frac{1}{2}(\mathsf{u}_0^{n-1} + \mathsf{u}_2^{n-1}) + \frac{(1-s)}{2\lambda}(\mathsf{v}_0^{n-1} - \mathsf{v}_2^{n-1}) + \frac{s}{2\lambda}(\mathsf{v}_0^{\mathrm{eq},n-1} - \mathsf{v}_2^{\mathrm{eq},n-1}). \tag{12.17}$$

Considering $-(\beta^+ - \beta^-)$(12.16) + (12.17) provides

$$\begin{aligned}
-(\beta^+ - \beta^-)\mathsf{u}_0^n + \mathsf{u}_1^n = &-\frac{1}{2}\Big(((\beta^+)^2 - (\beta^-)^2 - 1)\mathsf{u}_0^{n-1} + (\beta^+ - \beta^-)\mathsf{u}_1^{n-1} - \mathsf{u}_2^{n-1}\Big)\\
&-\frac{(1-s)}{2\lambda}\Big(\underbrace{((\beta^+ - \beta^-)^2 - 1)}_{=0}\mathsf{v}_0^{n-1} - (\beta^+ - \beta^-)\mathsf{v}_1^{n-1} + \mathsf{v}_2^{n-1}\Big)\\
&-\frac{s}{2\lambda}\Big(((\beta^+ - \beta^-)^2 - 1)\mathsf{v}_0^{\mathrm{eq},n-1} - (\beta^+ - \beta^-)\mathsf{v}_1^{\mathrm{eq},n-1} + \mathsf{v}_2^{\mathrm{eq},n-1}\Big) - (\beta^+ - \beta^-)\mathsf{S}_0^n,
\end{aligned}$$

thus

$$\begin{aligned}
\frac{(1-s)}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_1^{n-1} - \mathsf{v}_2^{n-1}\Big) = &-\lambda(\beta^+ - \beta^-)\mathsf{u}_0^n + \mathsf{u}_1^n + \frac{\lambda}{2}\Big(((\beta^+)^2 - (\beta^-)^2 - 1)\mathsf{u}_0^{n-1} + (\beta^+ - \beta^-)\mathsf{u}_1^{n-1} - \mathsf{u}_2^{n-1}\Big)\\
&-\frac{s}{2}\Big(((\beta^+ - \beta^-)\mathsf{v}_1^{\mathrm{eq},n-1} - \mathsf{v}_2^{\mathrm{eq},n-1}\Big) + \lambda(\beta^+ - \beta^-)\mathsf{S}_0^n.
\end{aligned}$$

Injecting this into (12.15) gives

$$(\beta^+ - \beta^-)\mathsf{v}_0^n - \mathsf{v}_1^n = -\lambda(\beta^+ - \beta^-)\mathsf{u}_0^n + \lambda\mathsf{u}_1^n + \lambda((\beta^+)^2 - (\beta^-)^2 - 1)\mathsf{u}_0^{n-1} + 2\lambda(\beta^+ - \beta^-)\mathsf{S}_0^n. \tag{12.18}$$

Used into (12.8), it yields the boundary scheme.

$$\mathsf{u}_0^{n+1} = \frac{1}{2}(s\beta^+ + (2-s)\beta^-)\mathsf{u}_0^n + \frac{1}{2}(2-s)\mathsf{u}_1^n + \frac{(1-s)}{2}((\beta^+)^2 - (\beta^-)^2 - 1)\mathsf{u}_0^{n-1} \tag{12.19}$$

$$+ \frac{s}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n} - \mathsf{v}_1^{\mathrm{eq},n}\Big) + \mathsf{S}_0^{n+1} + (1-s)(\beta^+ - \beta^-)\mathsf{S}_0^n. \tag{12.20}$$

We now check that the equation at $j=1$ is the bulk one. We write (12.10) for $j=1$, yielding

$$\mathsf{u}_1^{n+1} = \frac{1}{2}(\mathsf{u}_0^n + \mathsf{u}_2^n) + \frac{(1-s)}{2\lambda}(\mathsf{v}_0^n - \mathsf{v}_2^n) + \frac{s}{2\lambda}(\mathsf{v}_0^{\mathrm{eq},n} - \mathsf{v}_2^{\mathrm{eq},n}). \tag{12.21}$$

Writing (12.9) and (12.11) for $j=2$ at the previous time step and taking the difference gives

$$\begin{aligned}
\mathsf{v}_0^n - \mathsf{v}_2^n = &\frac{\lambda}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^{n-1} - 2\mathsf{u}_1^{n-1} + \mathsf{u}_3^{n-1}\Big) + \frac{(1-s)}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{n-1} - \mathsf{v}_3^{n-1}\Big)\\
&+\frac{s}{2}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n-1} - \mathsf{v}_3^{\mathrm{eq},n-1}\Big) + \lambda\mathsf{S}_0^n
\end{aligned} \tag{12.22}$$

We write (12.8) and (12.10) for $j=2$ at the previous time

$$\mathsf{u}_0^n = \frac{1}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^{n-1} + \mathsf{u}_1^{n-1}\Big) + \frac{(1-s)}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{n-1} - \mathsf{v}_1^{n-1}\Big) + \frac{s}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n-1} - \mathsf{v}_1^{\mathrm{eq},n-1}\Big) + \mathsf{S}_0^n \tag{12.23}$$

$$\mathsf{u}_2^n = \frac{1}{2}(\mathsf{u}_1^{n-1} + \mathsf{u}_3^{n-1}) + \frac{(1-s)}{2\lambda}(\mathsf{v}_1^{n-1} - \mathsf{v}_3^{n-1}) + \frac{s}{2\lambda}(\mathsf{v}_1^{\mathrm{eq},n-1} - \mathsf{v}_3^{\mathrm{eq},n-1}), \tag{12.24}$$

and we sum these two equations

$$\begin{aligned}
\mathsf{u}_0^n + \mathsf{u}_2^n = &\frac{1}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^{n-1} + 2\mathsf{u}_1^{n-1} + \mathsf{u}_3^{n-1}\Big) + \frac{(1-s)}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{n-1} - \mathsf{v}_3^{n-1}\Big)\\
&+\frac{s}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{\mathrm{eq},n-1} - \mathsf{v}_3^{\mathrm{eq},n-1}\Big) + \mathsf{S}_0^n.
\end{aligned} \tag{12.25}$$

Isolating the interesting term

$$\frac{(1-s)}{2}\Big((\beta^+-\beta^-)\mathsf{v}_0^{n-1}-\mathsf{v}_3^{n-1}\Big)=\lambda(\mathsf{u}_0^n+\mathsf{u}_2^n)-\frac{\lambda}{2}\Big((\beta^++\beta^-)\mathsf{u}_0^{n-1}+2\mathsf{u}_1^{n-1}+\mathsf{u}_3^{n-1}\Big)$$
$$-\frac{s}{2}\Big((\beta^+-\beta^-)\mathsf{v}_0^{\mathrm{eq},n-1}-\mathsf{v}_3^{\mathrm{eq},n-1}\Big)-\lambda\mathsf{S}_0^n. \quad (12.26)$$

Using this into (12.22) provides

$$\mathsf{v}_0^n-\mathsf{v}_2^n=\lambda(\mathsf{u}_0^n+\mathsf{u}_2^n)-2\lambda\mathsf{u}_1^{n-1},$$

which finally provides

$$\mathsf{u}_1^{n+1}=\frac{1}{2}(2-s)(\mathsf{u}_0^n+\mathsf{u}_2^n)-(1-s)\mathsf{u}_1^{n-1}+\frac{s}{2\lambda}(\mathsf{v}_0^{\mathrm{eq},n}-\mathsf{v}_2^{\mathrm{eq},n}), \quad (12.27)$$

coinciding with the bulk scheme. □

12.2.1.2 Consistency

Now that the non-conserved moment v has been eliminated, *cf.* Proposition 12.2.1, we can analyze the consistency of the boundary conditions at hand by using Taylor expansions on (12.5).

Proposition 12.2.2: Consistency of the boundary conditions

Under the condition

$$\beta^+-\beta^-=\pm1, \qquad \text{hence} \qquad \beta^-=\beta^+\mp1,$$

and for the acoustic scaling, the corresponding boundary Finite Difference scheme (12.5) for the $\mathrm{D_1Q_2}$ scheme (11.3) and (11.4) endowed with the boundary condition (12.2) has modified equation

$$\Big(-1+\frac{1}{2}(2\beta^+\mp(2-s))+\frac{1}{2}(2-s)+(1-s)(\pm\beta^+-1)\Big)u+\Big(1\pm(1-s)\Big)S+\frac{s}{2\lambda}\Big(\pm1-1\Big)v^{\mathrm{eq}}(u)$$
$$+\frac{\Delta x}{\lambda}\Big(-1-(1-s)(\pm\beta^+-1)\Big)\partial_t u+\Delta x\Big(\frac{1}{2}(2-s)-\frac{s}{2\lambda}\mathrm{d}v^{\mathrm{eq}}(u)\Big)\partial_x u+\frac{\Delta x}{\lambda}\partial_t S=O(\Delta x^2),$$

where the expansion has been computed around $(t^n,0)$ and we assumed that the source term S_0^{n+1} stems from a smooth function S.

Proof. As usual, one applies the boundary scheme (12.5) to smooth functions and perform Taylor expansions. □

Assume that the equilibrium is linear, so that $v^{\mathrm{eq}}(\mathsf{u})=V\mathsf{u}$. For the four conditions in Table 12.1, we obtain:

- **o-th order extrapolation**, we obtain

$$\frac{\lambda(s-2)}{\Delta x}S+\partial_t u+\frac{1}{2}(sV+\lambda(s-2))\partial_x u=O(\Delta x). \quad (12.28)$$

This suggests that one necessarily has to take $S\equiv0$, otherwise there could be some form of incompatibility. Assume for the moment $S\equiv0$. For $s\neq2$, we see that this is not equal to the transport equation inside the domain, thus the wave generated at the boundary does not have the same speed. Still, for $V<0$, we see that the velocity of this wave is negative for $s\in]0,2]$, thus this condition provides a good manner of having a transparent boundary condition, because it allows waves to exit from the domain at the outflow.

If we now consider $S=\Delta x\tilde{S}$ where $\tilde{S}=O(1)$ and plug the modified equation for the bulk scheme, namely

$$\partial_t u+V\partial_x u=O(\Delta x),$$

inside (12.28) to eliminate the time derivative, we obtain, under the assumption $s\neq2$

$$-\partial_x u=\frac{2\lambda}{\lambda+V}\tilde{S}+O(\Delta x),$$

hence this boundary condition is consistent with a non-homogeneous Neumann boundary condition.

- **Anti-bounce-back**
$$-(2-s)u+(2-s)S=O(\Delta x),$$
hence it gives, at leading order, the Dirichlet boundary condition $u=S$, which can be used as an inflow condition when $V>0$.
- **o-th order anti extrapolation**
$$-su+sS-\frac{s}{\lambda}Vu=O(\Delta x).$$
At leading order, this corresponds to the Dirichlet boundary condition $u=S/(1+V/\lambda)$.
- **Bounce-back**
$$sS-\frac{s}{\lambda}Vu=O(\Delta x).$$
This is the Dirichlet boundary condition $u=\lambda/VS$.

12.2.1.3 Consistency without corresponding Finite Difference scheme: Maxwell iteration

The previous consistency analysis has been possible because we have succeeded in rewriting everything solely on the conserved moment. Inspired by the Maxwell iteration [Yong et al., 2016] introduced in Section 8.5.2, we try to develop a formal analysis which does not need Proposition 12.2.1.

The analysis that we develop is totally formal and we shall observe that it yields the same results as Section 12.2.1.2. Still, many points of this procedure (for example, the fact that spatial derivatives of the boundary source term S can appear) are not fully understood.

We first write the analogue at the boundary of the matrices $\mathbf{A}$ and $\mathbf{B}$ introduced in Chapter 7.

$$\mathbf{A}_{\partial\Omega}=\begin{bmatrix}\frac{(1-s_1)}{2}\Big((\beta^+-\beta^-)+\bar{\mathrm{x}}\Big) & \frac{(1-s)}{2\lambda}\Big((\beta^+-\beta^-)-\bar{\mathrm{x}}\Big)\\ \frac{\lambda(1-s_1)}{2}\Big((\beta^+-\beta^-)-\bar{\mathrm{x}}\Big) & \frac{(1-s)}{2}\Big((\beta^+-\beta^-)+\bar{\mathrm{x}}\Big)\end{bmatrix},\ \mathbf{B}_{\partial\Omega}=\begin{bmatrix}\frac{s_1}{2}\Big((\beta^+-\beta^-)+\bar{\mathrm{x}}\Big) & \frac{s}{2\lambda}\Big((\beta^+-\beta^-)-\bar{\mathrm{x}}\Big)\\ \frac{\lambda s_1}{2}\Big((\beta^+-\beta^-)-\bar{\mathrm{x}}\Big) & \frac{s}{2}\Big((\beta^+-\beta^-)+\bar{\mathrm{x}}\Big)\end{bmatrix},$$

where we recall—*cf.* Section 8.5.2—that the Maxwell iteration obliges us to consider some relaxation parameter s_1 also for the conserved moment. Their asymptotic equivalent are

$$\mathbf{A}_{\partial\Omega}\asymp\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}+\Delta x\boldsymbol{\mathcal{A}}^{(1)}_{\partial\Omega}+O(\Delta x^2),\qquad \mathbf{B}_{\partial\Omega}\asymp\boldsymbol{\mathcal{B}}^{(0)}_{\partial\Omega}+\Delta x\boldsymbol{\mathcal{B}}^{(1)}_{\partial\Omega}+O(\Delta x^2)$$

with

$$\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}=\begin{bmatrix}\frac{(1-s_1)}{2}\Big((\beta^+-\beta^-)+1\Big) & \frac{(1-s)}{2\lambda}\Big((\beta^+-\beta^-)-1\Big)\\ \frac{\lambda(1-s_1)}{2}\Big((\beta^+-\beta^-)-1\Big) & \frac{(1-s)}{2}\Big((\beta^+-\beta^-)+1\Big)\end{bmatrix},\qquad \boldsymbol{\mathcal{A}}^{(1)}_{\partial\Omega}=\begin{bmatrix}\frac{(1-s_1)}{2} & -\frac{(1-s)}{2\lambda}\\ -\frac{\lambda(1-s_1)}{2} & \frac{(1-s)}{2}\end{bmatrix}\partial_x,$$
$$\boldsymbol{\mathcal{B}}^{(0)}_{\partial\Omega}=\begin{bmatrix}\frac{s_1}{2}\Big((\beta^+-\beta^-)+1\Big) & \frac{s}{2\lambda}\Big((\beta^+-\beta^-)-1\Big)\\ \frac{\lambda s_1}{2}\Big((\beta^+-\beta^-)-1\Big) & \frac{s}{2}\Big((\beta^+-\beta^-)+1\Big)\end{bmatrix},\qquad \boldsymbol{\mathcal{B}}^{(1)}_{\partial\Omega}=\begin{bmatrix}\frac{s_1}{2} & -\frac{s}{2\lambda}\\ -\frac{\lambda s_1}{2} & \frac{s}{2}\end{bmatrix}\partial_x.$$

Observe that the latter matrix does not depend on β^- and β^+ because we are analyzing local boundary conditions.

- **o-th order extrapolation**. We have to check if the formal series which is asymptotically equivalent to $z\boldsymbol{I}-\mathbf{A}_{\partial\Omega}$ is a unit. This boils down to ensuring
$$\zeta^{(0)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}=\begin{bmatrix}s_1 & 0\\ 0 & s\end{bmatrix},\quad\text{hence}\quad\det(\zeta^{(0)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega})=s_1s,$$
thus the matrix is invertible for $s_1,s\neq0$. Under this condition
$$\begin{aligned}&\Big(\zeta^{(0)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}+\Delta x(\zeta^{(1)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(1)}_{\partial\Omega})+O(\Delta x^2)\Big)^{-1}\\ &=\Big(\zeta^{(0)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}\Big)^{-1}-\Delta x\Big(\zeta^{(0)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}\Big)^{-1}(\zeta^{(1)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(1)}_{\partial\Omega})\Big(\zeta^{(0)}\boldsymbol{I}-\boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}\Big)^{-1}+O(\Delta x^2)\\ &=\begin{bmatrix}\frac{1}{s_1} & 0\\ 0 & \frac{1}{s}\end{bmatrix}-\Delta x\begin{bmatrix}\frac{1}{\lambda s_1^2}\partial_t-\frac{(1-s_1)}{2s_1^2}\partial_x & \frac{(1-s)}{2\lambda s_1 s}\partial_x\\ \frac{\lambda(1-s_1)}{2s_1 s}\partial_x & \frac{1}{\lambda s^2}\partial_t-\frac{(1-s)}{2s^2}\partial_x\end{bmatrix}+O(\Delta x^2).\end{aligned}$$

We then have

$$\begin{aligned}
&\Big(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} + \Delta x(\zeta^{(1)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(1)}_{\partial\Omega}) + O(\Delta x^2)\Big)^{-1}\Big(\boldsymbol{\mathcal{B}}^{(0)}_{\partial\Omega} + \Delta x\boldsymbol{\mathcal{B}}^{(1)}_{\partial\Omega} + O(\Delta x^2)\Big)\\
&= \boldsymbol{I} - \Delta x\begin{bmatrix} \frac{1}{\lambda s_1}\partial_t - \frac{(1-s_1)}{2s_1}\partial_x & \frac{(1-s)}{2\lambda s_1}\partial_x \\ \frac{\lambda(1-s_1)}{2s}\partial_x & \frac{1}{\lambda s}\partial_t - \frac{(1-s)}{2s}\partial_x \end{bmatrix} + \Delta x\begin{bmatrix} \frac{1}{2}\partial_x & -\frac{s}{2\lambda s_1}\partial_x \\ -\frac{\lambda s_1}{2s}\partial_x & \frac{1}{2}\partial_x \end{bmatrix} + O(\Delta x^2)\\
&= \boldsymbol{I} + \Delta x\begin{bmatrix} -\frac{1}{\lambda s_1}\partial_t + \frac{1}{2s_1}\partial_x & -\frac{1}{2\lambda s_1}\partial_x \\ \star & \star \end{bmatrix} + O(\Delta x^2)
\end{aligned}$$

This provides, if we take $S \equiv 0$

$$\partial_t u - \frac{\lambda}{2}\partial_x u + \frac{V}{2}\partial_x u = O(\Delta x).$$

This equation looks different from the one obtained by the Finite Difference scheme but we have, multiplying by $2 - s$

$$(2-s)\partial_t u - \frac{\lambda(2-s)}{2}\partial_x u + \frac{V(2-s)}{2}\partial_x u = O(\Delta x),$$

and doing some basic algebraic manipulations

$$(2-s)\partial_t u - \frac{\lambda(2-s)}{2}\partial_x u + \underbrace{\frac{V(2-s)}{2}\partial_x u - \frac{sV}{2}\partial_x u}_{= V(1-s)\partial_x u = -(1-s)\partial_t u + O(\Delta x)} + \frac{sV}{2}\partial_x u = O(\Delta x),$$

using the bulk modified equation. This gives the expected equation

$$\partial_t u + \frac{1}{2}\Big(sV - \lambda(2-s)\Big)\partial_x u = O(\Delta x),$$

which is the same as if we use the corresponding Finite Difference scheme at the boundary, *cf.* Proposition 12.2.2.

- **Anti-bounce-back**. Again, we check for invertibility:

$$\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} = \begin{bmatrix} 1 & 0 \\ \lambda(1-s_1) & s \end{bmatrix}, \quad \text{hence} \quad \det(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}) = s,$$

thus the matrix is invertible for $s \neq 0$. In this case, having

$$\Big(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} + O(\Delta x)\Big)^{-1} = \begin{bmatrix} 1 & 0 \\ -\frac{\lambda(1-s_1)}{s} & \frac{1}{s} \end{bmatrix} + O(\Delta x), \quad \boldsymbol{\mathcal{B}}^{(0)}_{\partial\Omega} + O(\Delta x) = \begin{bmatrix} 0 & 0 \\ -\lambda s_1 & s \end{bmatrix} + O(\Delta x),$$

we are left with

$$u = S + O(\Delta x).$$

- **o-th order anti extrapolation**. We have

$$\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} = \begin{bmatrix} 1 & \frac{(1-s)}{\lambda} \\ \lambda(1-s_1) & 1 \end{bmatrix}, \quad \text{hence} \quad \det(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}) = s_1 + s - s_1 s,$$

thus the matrix is invertible since we can always select s_1 in order to make this true. In particular, $s_1 = 0$ does the job.

$$\Big(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} + O(\Delta x)\Big)^{-1} = \frac{1}{(s_1 + s - s_1 s)}\begin{bmatrix} 1 & -\frac{(1-s)}{\lambda} \\ -\lambda(1-s_1) & 1 \end{bmatrix} + O(\Delta x)$$

$$\boldsymbol{\mathcal{B}}^{(0)}_{\partial\Omega} + O(\Delta x) = \begin{bmatrix} 0 & -\frac{s}{\lambda} \\ -\lambda s_1 & 0 \end{bmatrix} + O(\Delta x).$$

Taking the boundary source term into account provides

$$u = \frac{s_1(1-s)}{(s_1+s-s_1 s)}u - \frac{sV}{\lambda(s_1+s-s_1 s)}u + \frac{s}{(s_1+s-s_1 s)}S + O(\Delta x),$$

which after manipulations yields

$$u = S/(1+V/\lambda) + O(\Delta x).$$

- **Bounce-back**. We gain

$$\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} = \begin{bmatrix} s_1 & \frac{(1-s)}{\lambda} \\ 0 & 1 \end{bmatrix}, \quad \text{hence} \quad \det(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega}) = s_1,$$

thus the matrix is invertible for $s_1 \neq 0$.

$$\left(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}^{(0)}_{\partial\Omega} + O(\Delta x)\right)^{-1} = \begin{bmatrix} \frac{1}{s_1} & -\frac{(1-s)}{\lambda s_1} \\ 0 & 1 \end{bmatrix} + O(\Delta x), \qquad \boldsymbol{\mathcal{B}}^{(0)}_{\partial\Omega} + O(\Delta x) = \begin{bmatrix} s_1 & -\frac{s}{\lambda} \\ 0 & 0 \end{bmatrix} + O(\Delta x).$$

Therefore

$$u = u - \frac{sV}{\lambda s_1}u + \frac{s}{s_1}S + O(\Delta x),$$

thus

$$u = \lambda/VS.$$

We see that for each condition in Table 12.1, our procedure based on the Maxwell iteration yields the same results, at leading order, as the consistency analysis that we have done in Section 12.2.1.2 thanks to Proposition 12.2.1. This suggests that this formal procedure can yield important information on the behavior of the boundary conditions. It can be resumed as follows:

1. Consider

$$(\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}_{\partial\Omega})\boldsymbol{m} = \boldsymbol{\mathcal{B}}_{\partial\Omega}\boldsymbol{m}^{\mathrm{eq}} + \boldsymbol{S},$$

   and invert the formal series on the left hand side (up to the desired order) to obtain

$$\boldsymbol{m} = (\zeta\boldsymbol{I} - \boldsymbol{\mathcal{A}}_{\partial\Omega})^{-1}(\boldsymbol{\mathcal{B}}_{\partial\Omega}\boldsymbol{m}^{\mathrm{eq}} + \boldsymbol{S}),$$

   from which select the desired equation.

2. Use the modified equation of the bulk scheme to further elaborate the terms.

All the results concerning consistency for these boundary conditions have been assessed both in the linear and non-linear context and the theoretical prediction are totally confirmed by numerical experiments, which are not reported in this manuscript.

12.2.2 A non-local boundary condition: first order extrapolation

We have seen that the o-th order extrapolation allows to propose transparent boundary conditions, which are especially useful when $V < 0$: the boundary is an outflow. Now, instead of considering local boundary conditions like (12.2), we replace $\mathsf{f}^{+,n,\star}_{-1}$ by its first order extrapolation. This is proposed in the spirit of [Guo et al., 2002] and [LeVeque, 2002, Equation (7.4)]) and gives

$$\mathsf{f}^{+,n+1}_0 = 2\mathsf{f}^{+,n,\star}_0 - \mathsf{f}^{+,n,\star}_1 \tag{12.29}$$

$$\mathsf{f}^{-,n+1}_0 = \mathsf{f}^{-,n,\star}_1, \tag{12.30}$$

where we do not consider a source term.

12.2.2.1 Elimination of the non-conserved moment

We can indeed eliminate the non-conserved moment v, as stated by the following result.

Proposition 12.2.3: Corresponding Finite Difference scheme

The corresponding Finite Difference scheme for the D_1Q_2 scheme (11.3) and (11.4) endowed with the first order extrapolation boundary condition (12.29) is

$$\begin{aligned} u_j^{n+1} &= \frac{1}{2}(2-s)(u_{j-1}^n + u_{j+1}^n) - (1-s)u_j^{n-1} + \frac{s}{2\lambda}(v_{j-1}^{\mathrm{eq},n} - v_{j+1}^{\mathrm{eq},n}), \qquad j \geq 1, \\ u_0^{n+1} &= u_0^n + (1-s)u_1^n - (1-s)u_1^{n-1} + \frac{s}{\lambda}(v_0^{\mathrm{eq},n} - v_1^{\mathrm{eq},n}). \end{aligned} \tag{12.31}$$

Observe that contrarily to the o-th order extrapolation boundary condition, the scheme at the boundary (12.31) is genuinely multi-step for $s \neq 1$, like the bulk scheme. We observe that this boundary scheme does not look as—to the best of our knowledge—any standard scheme that can be found in textbooks, except for $s = 1$ where it coincides with the Lax-Friedrichs scheme with second-order extrapolation, see [LeVeque, 2002].

Proof of Proposition 12.2.3. Taking the collision model into account

$$f_0^{+,n+1} = u_0^n + \frac{(1-s)}{\lambda}v_0^n + \frac{s}{\lambda}v_0^{\mathrm{eq},n} - \frac{1}{2}u_1^n - \frac{(1-s)}{2\lambda}v_1^n - \frac{s}{2\lambda}v_1^{\mathrm{eq},n}, \tag{12.32}$$

$$f_0^{-,n+1} = \frac{1}{2}u_1^n - \frac{(1-s)}{2\lambda}v_1^n - \frac{s}{2\lambda}v_1^{\mathrm{eq},n}. \tag{12.33}$$

Recasting on the moments yields

$$u_0^{n+1} = u_0^n + \frac{(1-s)}{\lambda}(v_0^n - v_1^n) + \frac{s}{\lambda}(v_0^{\mathrm{eq},n} - v_1^{\mathrm{eq},n}), \tag{12.34}$$

$$v_0^{n+1} = \lambda(u_0^n - u_1^n) + (1-s)v_0^n + sv_0^{\mathrm{eq},n}. \tag{12.35}$$

Let us start by writing the Finite Difference scheme at $j = 0$. Writing (12.34) and (12.11) for $j = 1$ at the previous time and taking the difference gives

$$v_0^n - v_1^n = \frac{\lambda}{2}(u_0^{n-1} - 2u_1^{n-1} + u_2^{n-1}) + \frac{(1-s)}{2}(v_0^{n-1} - v_2^{n-1}) + \frac{s}{2}(v_0^{\mathrm{eq},n-1} - v_2^{\mathrm{eq},n-1}). \tag{12.36}$$

Writing (12.10) at $j = 1$ at the previous time step is enough and provides

$$u_1^n = \frac{1}{2}(u_0^{n-1} + u_2^{n-1}) + \frac{(1-s)}{2\lambda}(v_0^{n-1} - v_2^{n-1}) + \frac{s}{2\lambda}(v_0^{\mathrm{eq},n-1} - v_2^{\mathrm{eq},n-1}), \tag{12.37}$$

thus isolating the desired term

$$\frac{(1-s)}{2}(v_0^{n-1} - v_2^{n-1}) = \lambda u_1^n - \frac{\lambda}{2}(u_0^{n-1} + u_2^{n-1}) - \frac{s}{2}(v_0^{\mathrm{eq},n-1} - v_2^{\mathrm{eq},n-1}), \tag{12.38}$$

and plugging back into (12.36) gives

$$v_0^n - v_1^n = \lambda u_1^n - \lambda u_1^{n-1}, \tag{12.39}$$

hence finally inside

$$u_0^{n+1} = u_0^n + (1-s)u_1^n - (1-s)u_1^{n-1} + \frac{s}{\lambda}(v_0^{\mathrm{eq},n} - v_1^{\mathrm{eq},n}). \tag{12.40}$$

Let us now analyze the scheme at $j = 1$, to check that it equals the bulk scheme. We have to consider (12.10) for $j = 1$:

$$u_1^{n+1} = \frac{1}{2}(u_0^n + u_2^n) + \frac{(1-s)}{2\lambda}(v_0^n - v_2^n) + \frac{s}{2\lambda}(v_0^{\mathrm{eq},n} - v_2^{\mathrm{eq},n}). \tag{12.41}$$

Using (12.35) and (12.11) for $j=2$ at the previous time and taking the difference

$$v_0^n - v_2^n = \frac{\lambda}{2}(2u_0^{n-1} - 3u_1^{n-1} + u_3^{n-1}) + \frac{(1-s)}{2}(2v_0^{n-1} - v_1^{n-1} - v_3^{n-1}) + \frac{s}{2}(2v_0^{\mathrm{eq},n-1} - v_1^{\mathrm{eq},n-1} - v_3^{\mathrm{eq},n-1}). \tag{12.42}$$

Writing (12.34) and (12.10) for $j=2$ at the previous time and summing provides

$$u_0^n + u_2^n = \frac{1}{2}(2u_0^{n-1} + u_1^{n-1} + u_3^{n-1}) + \frac{(1-s)}{2\lambda}(2v_0^{n-1} - v_1^{n-1} - v_3^{n-1}) + \frac{s}{2\lambda}(2v_0^{\mathrm{eq},n-1} - v_1^{\mathrm{eq},n-1} - v_3^{\mathrm{eq},n-1}). \tag{12.43}$$

By isolating the interesting term

$$\frac{(1-s)}{2}(2v_0^{n-1} - v_1^{n-1} - v_3^{n-1}) = \lambda(u_0^n + u_2^n) - \frac{\lambda}{2}(2u_0^{n-1} + u_1^{n-1} + u_3^{n-1}) - \frac{s}{2}(2v_0^{\mathrm{eq},n-1} - v_1^{\mathrm{eq},n-1} - v_3^{\mathrm{eq},n-1}). \tag{12.44}$$

Inserting into (12.42) yields

$$v_0^n - v_2^n = \lambda(u_0^n + u_2^n) - 2\lambda u_1^{n-1}. \tag{12.45}$$

Finally putting into (12.41)

$$u_1^{n+1} = \frac{1}{2}(2-s)(u_0^n + u_2^n) - (1-s)u_1^{n-1} + \frac{s}{2\lambda}(v_0^{\mathrm{eq},n} - v_2^{\mathrm{eq},n}), \tag{12.46}$$

coinciding with the bulk scheme. □

12.2.2.2 Consistency

Under the acoustic scaling, we obtain, using Taylor expansions at the boundary, that this boundary condition is consistent with

$$\partial_t u + \partial_x v^{\mathrm{eq}}(u) = O(\Delta x).$$

This is exactly the same equation as the bulk scheme, which means that the condition is suitable to be used as transparent boundary condition when—in the linear case—$V<0$, since the wave from the scheme at the boundary has precisely the same velocity as the one for the bulk scheme.

12.2.2.3 Consistency without corresponding Finite Difference scheme: Maxwell iteration

Again, we can check consistency by the help of the Maxwell iteration. The condition to check is

$$\det(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}_{\partial\Omega}^{(0)}) = s_1 s \neq 0,$$

which is satisfied using $s_1 \neq 0$. This gives, through the usual computations

$$\left(\zeta^{(0)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}_{\partial\Omega}^{(0)} + \Delta x(\zeta^{(1)}\boldsymbol{I} - \boldsymbol{\mathcal{A}}_{\partial\Omega}^{(1)}) + O(\Delta x^2)\right)^{-1}(\boldsymbol{\mathcal{B}}_{\partial\Omega}^{(0)} + \Delta x\boldsymbol{\mathcal{B}}_{\partial\Omega}^{(1)} + O(\Delta x^2)) = \boldsymbol{I} - \Delta x\begin{bmatrix} \frac{1}{\lambda s_1}\partial_t & \frac{1}{\lambda s_1}\partial_x \\ \star & \star \end{bmatrix} + O(\Delta x^2).$$

Therefore:

$$u = u - \frac{\Delta x}{\lambda s_1}(\partial_t u + \partial_x v^{\mathrm{eq}}) + O(\Delta x^2), \tag{12.47}$$

This confirms once more that we can use the Maxwell iteration to probe the consistency of the boundary conditions.

12.2.3 Stability

We now want to study the stability of the different boundary conditions that we have introduced in Section 12.2.1 and Section 12.2.2. To do this, we try to use the so-called "GKS theory" [Gustafsson et al., 1972]. We do not aim at providing a full and rigorous introduction to this theory, which can be found in [Strikwerda, 2004, Chapter 11],

[Gustafsson et al., 1995, Chapter 13] and [Coulombel, 2011a]. We do not even precisely define the concept of GKS stability, which is pretty involved.

In Section 12.2.3, since we are interested in linear stability for the boundary conditions, we consider a linear collision operator where $v^{\mathrm{eq}}(u) = Vu$. Moreover, we take $s = 2$, so that the bulk scheme is the leap-frog scheme:

$$u_j^{n+1} = u_j^{n-1} + \frac{V}{\lambda}(u_{j-1}^n - u_{j+1}^n), \qquad j \geq 1, \tag{12.48}$$

which is a non-dissipative (dispersive) scheme, thus provides the setting where instabilities can indeed develop. We observe that dissipation rules out mild (or borderline) instabilities, see [Trefethen, 1984], that we will better define. In order to deal with an outflow, we assume the CFL condition $-1 < V/\lambda < 0$.

12.2.3.1 Survival kit on the GKS theory

Introducing the forward time shift z, see Definition 8.1.1, the bulk equation (12.48) becomes

$$\left(z - \frac{1}{z}\right)\tilde{u}_j = \frac{V}{\lambda}(\tilde{u}_{j-1} - \tilde{u}_{j+1}), \tag{12.49}$$

where the tilde indicates that we have essentially applied a Laplace transform. It is customary—in order to obtain the so-called "characteristic equation" (*i.e.* the amplification polynomial equals zero)—to plug the *ansatz* $\tilde{u}_j = \kappa^j$, where $\kappa = \kappa(z)$ plays essentially the role of the basic shift operator $\bar{x} = x^{-1}$. We then obtain

$$\left(z - \frac{1}{z}\right) = -\frac{V}{\lambda}\left(\kappa - \frac{1}{\kappa}\right). \tag{12.50}$$

In the context of the study of boundary conditions, one has to interpret the time shift z as the unknown and the space shift κ as a function of this unknown. The solutions $\kappa_\pm = \kappa_\pm(z)$ of the characteristic equation (12.50) make up—by superposition principle—the solution that we look for, which reads

$$\tilde{u}_j = A_-(z)\kappa_-(z)^j + A_+(z)\kappa_+(z)^j, \qquad \text{or} \qquad u_j^n = a_- z^n \kappa_-(z)^j + a_+ z^n \kappa_+(z)^j, \tag{12.51}$$

provided that κ_- and κ_+ are distinct, where the coefficients of the linear combination have to be determined. Observe that in general, one is interested in the regime where $z \in \mathbb{C}$ and $|z| \geq 1$. The roots of the characteristic equation—like (12.50)—split into two groups for $|z| \geq 1$, *cf.* [Strikwerda, 2004, Theorem 11.3.1]

Theorem 12.2.1

Assume that the bulk scheme is stable with restricted *von Neumann* (*cf.* Theorem 7.7.3). Then, there exist two integer K_- and K_+ such that the roots $\kappa(z)$ of the characteristic equation are split into two groups:

$$\begin{aligned} &|\kappa_{-,r}(z)| < 1, \quad \text{for} \quad |z| > 1, \quad r \in [\![1, K_-]\!], \quad \text{(stable roots)}, \\ &|\kappa_{+,r}(z)| > 1, \quad \text{for} \quad |z| > 1, \quad r \in [\![1, K_+]\!], \quad \text{(unstable roots)}. \end{aligned}$$

In our case, where the bulk scheme is the leap-frog scheme, Theorem 12.2.1 is fulfilled with $K_- = K_+ = 1$, and easy computations show [Gustafsson et al., 1972, Lemma 6.2] and [Trefethen, 1984] that $\kappa_-(1) = -1 = -\kappa_+(1)$, $\kappa_-(-1) = 1 = -\kappa_+(-1)$ and indeed $\kappa_- = -1/\kappa_+$. Since we are looking for solutions which are L^2 bounded on the discretization $\Delta x\mathbb{Z}$ of the half line $\mathbb{R}_+$, we take solutions of the form

$$\tilde{u}_j = A_-(z)\kappa_-(z)^j, \tag{12.52}$$

because otherwise they would be growing at infinity for $|z| > 1$.

The conditions that we consider here are listed in Table 12.2. To study them in terms of stability, one takes their Laplace transform (adding a source which is however only needed to understand the keystone of this way

Table 12.2: Different boundary conditions that we consider for the leap-frog scheme with their references in the literature, if they stem from some boundary condition for the D_1Q_2 scheme with $s = 2$ (*cf.* Proposition 12.2.1 and Proposition 12.2.3) and if they are GKS stable.

	Condition	In the literature [Strikwerda, 2004]	[Gustafsson et al., 1995]	[Gustafsson et al., 1972]	[Trefethen, 1984]	[Trefethen, 1982]	LBM	GKS st.
1	$u_0^{n+1} = u_1^{n+1}$	(11.2.2a)	(13.1.38b) $q = 1$	(6.3a) $j = 1$	(2.5δ)	Table 5.1 δ	No	No
2	$u_0^{n+1} = u_1^n$	(11.2.2b)			(2.5α)	Table 5.1 α	No	Yes
3	$u_0^{n+1} = \frac{1}{2}(u_0^n + u_2^n)$					Table 5.1 γ	No	No
4	$u_0^{n+1} = 2u_1^{n+1} - u_2^{n+1}$	(11.2.15a)					No	No
5	$u_0^{n+1} = 2u_1^n - u_2^{n-1}$	(11.2.15b)	(13.1.60)				No	Yes
6	$u_0^{n+1} = u_0^n - \frac{V}{\lambda}(u_1^n - u_0^n)$	(11.2.2d)	(13.1.42)	(6.3b)			oth or. extrap.	Yes
7	$u_0^{n+1} = u_1^{n-1} - \left(1 + \frac{2V}{\lambda}\right)(u_1^n - u_0^n)$						1st. or. extrap.	No
8	$u_0^{n+1} = u_0^{n-1} - \frac{V}{\lambda}(u_1^n - u_0^n)$						ABB	No
9	$u_0^{n+1} = u_0^{n-1} - \frac{2V}{\lambda}(u_1^n - u_0^n)$	(12.2.2c)					No	No
10	$u_0^{n+1} = 2u_0^n - u_0^{n-1} - \frac{V}{\lambda}(u_1^n - u_0^n)$						$\beta^+ = 2, \beta^- = 1$	No

of proceeding). For example, for conditions (6), (7) and (10), we obtain

$$\begin{aligned}
&\text{Condition (6)} \qquad z\tilde{u}_0 = \tilde{u}_0 - \frac{V}{\lambda}(\tilde{u}_1 - \tilde{u}_0) + \tilde{S}(z),\\
&\text{Condition (7)} \qquad z\tilde{u}_0 = \frac{1}{z}\tilde{u}_1 - \left(1 + \frac{2V}{\lambda}\right)(\tilde{u}_1 - \tilde{u}_0) + \tilde{S}(z),\\
&\text{Condition (10)} \qquad z\tilde{u}_0 = 2\tilde{u}_0 - \frac{1}{z}\tilde{u}_0 - \frac{V}{\lambda}(\tilde{u}_1 - \tilde{u}_0) + \tilde{S}(z),
\end{aligned}$$

Then, one plugs the stable modal solution (12.52) into the Laplace transformed boundary condition in order to obtain an expression for the coefficient $A_-(z)$. For the example:

$$\begin{aligned}
&\text{Condition (6)} \qquad \left[z - 1 + \frac{V}{\lambda}(\kappa_-(z) - 1)\right] A_-(z) = \tilde{S}(z),\\
&\text{Condition (7)} \qquad \left[z - \frac{\kappa_-(z)}{z} + \left(1 + \frac{2V}{\lambda}(\kappa_-(z) - 1)\right)\right] A_-(z) = \tilde{S}(z),\\
&\text{Condition (10)} \qquad \left[z - 2 + \frac{1}{z} + \frac{V}{\lambda}(\kappa_-(z) - 1)\right] A_-(z) = \tilde{S}(z).
\end{aligned}$$

This need to be interpreted as a linear system where the right hand side $\tilde{S}(z)$ is the forcing driving term which shall determine $A_-(z)$ for every $|z| \geq 1$. For this reason, in order to deal with a well posed problem depending continuously on $\tilde{S}(z)$, the coefficients of $A_-(z)$ in this expressions do not have to vanish for $|z| \geq 1$. Otherwise, the scheme shall develop instabilities at the boundary. For they generally give rise to moderate (essentially linear in time) growth in the solution, when the stability condition is violated for $|z| = 1$, we shall say that the instability is "mild", whereas when this happens for $|z| > 1$, the effect are in general catastrophic and we call this instability "exponential" or of "Godunov-Ryabenkii"-type. Let us analyze the three conditions that we have selected for illustrative purpose.

- Condition (6). This has already been shown [Gustafsson et al., 1972, Strikwerda, 2004] to be stable, since it is a pretty standard transparent boundary condition. Nothing more is needed.
- Condition (7). Taking $z = -1$ on the unit circle, we obtain

$$\left[z - \frac{\kappa_-(z)}{z} + \left(1 + \frac{2V}{\lambda}(\kappa_-(z) - 1)\right)\right]_{z=-1} = 0,$$

  hence we have found an unstable mode on the unit circle which is $(z, \kappa_-, \kappa_+) = (-1, 1, -1)$. For this reason, the instability is indeed "mild", *cf.* the numerical simulations to come.
- Condition (10). The condition does not seem to give problem on the unit disk $|z| = 1$. In order to show that the unstable mode generated outside the unit disk and hence the instability is "exponential", we look for κ_- fulfilling

$$z - 2 + \frac{1}{z} + \frac{V}{\lambda}(\kappa_-(z) - 1) = 0, \qquad \text{hence} \qquad \kappa_-(z) = 1 - \frac{\lambda}{V}\left(z - 2 + \frac{1}{z}\right). \tag{12.53}$$

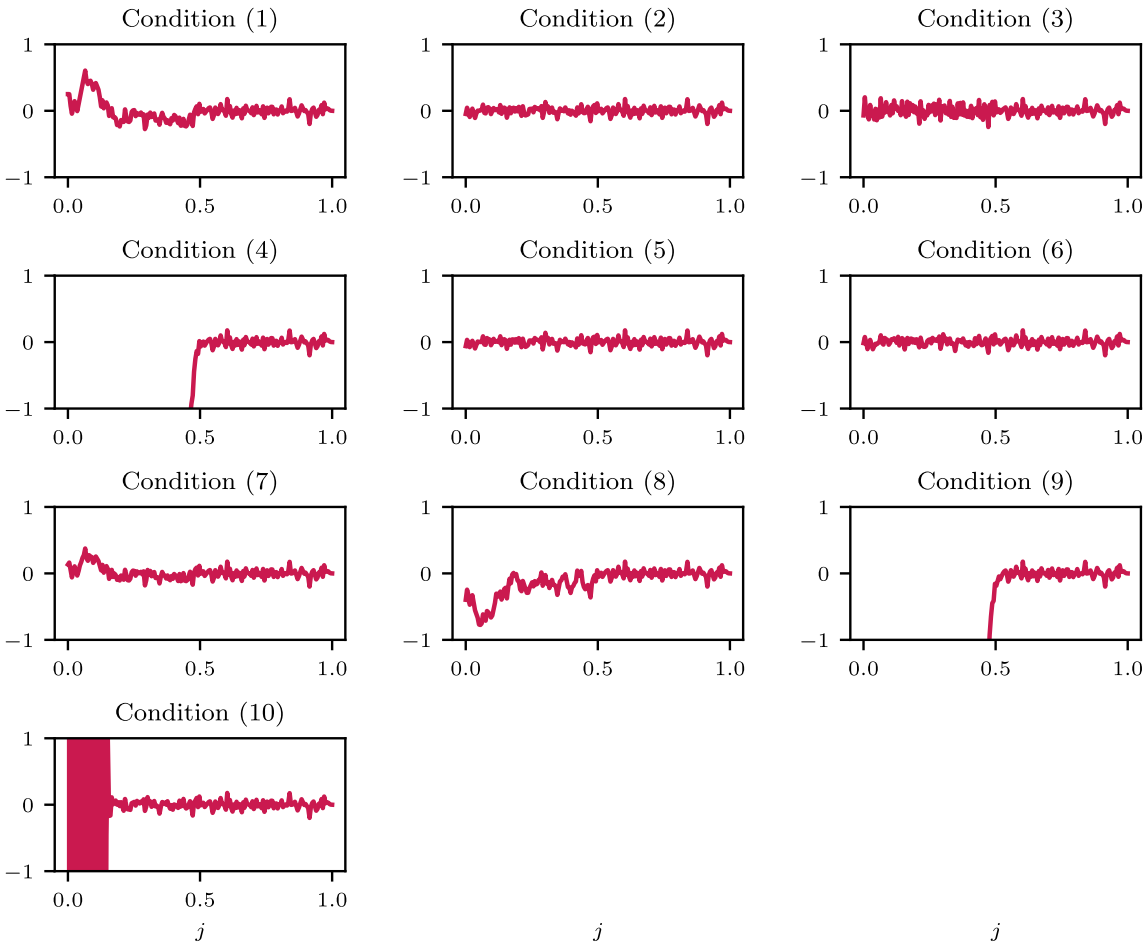


Figure 12.2: Solution of the leap-from scheme at the final time $T = 1$ for the conditions in Table 12.2.

Inserting this into the characteristic equation (12.50) gives

$$\left(z - \frac{1}{z}\right) = -\frac{V}{\lambda}\left(1 - \frac{\lambda}{V}\left(z - 2 + \frac{1}{z}\right) - \frac{1}{1 - \lambda/V(z - 2 + 1/z)}\right). \tag{12.54}$$

This is a fourth order equation in z. Consider the choice that we shall frequently adopt, that is $V/\lambda = -1/2$. In this case, the solutions are obtained by computer algebra and read

$$z = 1, \qquad z = \frac{5}{6} \pm \frac{\sqrt{23}}{6} i, \quad \text{with} \quad \left|\frac{5}{6} \pm \frac{\sqrt{23}}{6} i\right| = \frac{2}{\sqrt{3}} > 1.$$

We are not interested in $z = 1$, because for this value (12.53) is not fulfilled (it would be the case for κ_+). We have $\kappa_-(\frac{5}{6} \pm \frac{\sqrt{23}}{6} i) = -\frac{1}{12} \pm \frac{\sqrt{23}}{12} i$, which is inside the unit circle thus is really κ_- (and not κ_+). This is an example of Godunov-Ryabenkii eigensolution [Trefethen, 1984], which generates an exponential growth of the solution starting from the boundary.

12.2.3.2 Numerical study on some boundary conditions

Though the GKS theory—at least in its original formulation—only analyzes the case where the schemes are taken with zero initial data (which is in general never the case for multi-step schemes), a good empirical way of checking the stability of boundary conditions [Trefethen, 1982] in real situations is to consider random initial data, which are particularly rich of high-frequency harmonics. Here, we simulate using the leap-frog scheme (and not the lattice Boltzmann D_1Q_2 scheme), on the bounded domain $\Omega = [0, 1]$ endowed with Dirichlet boundary conditions

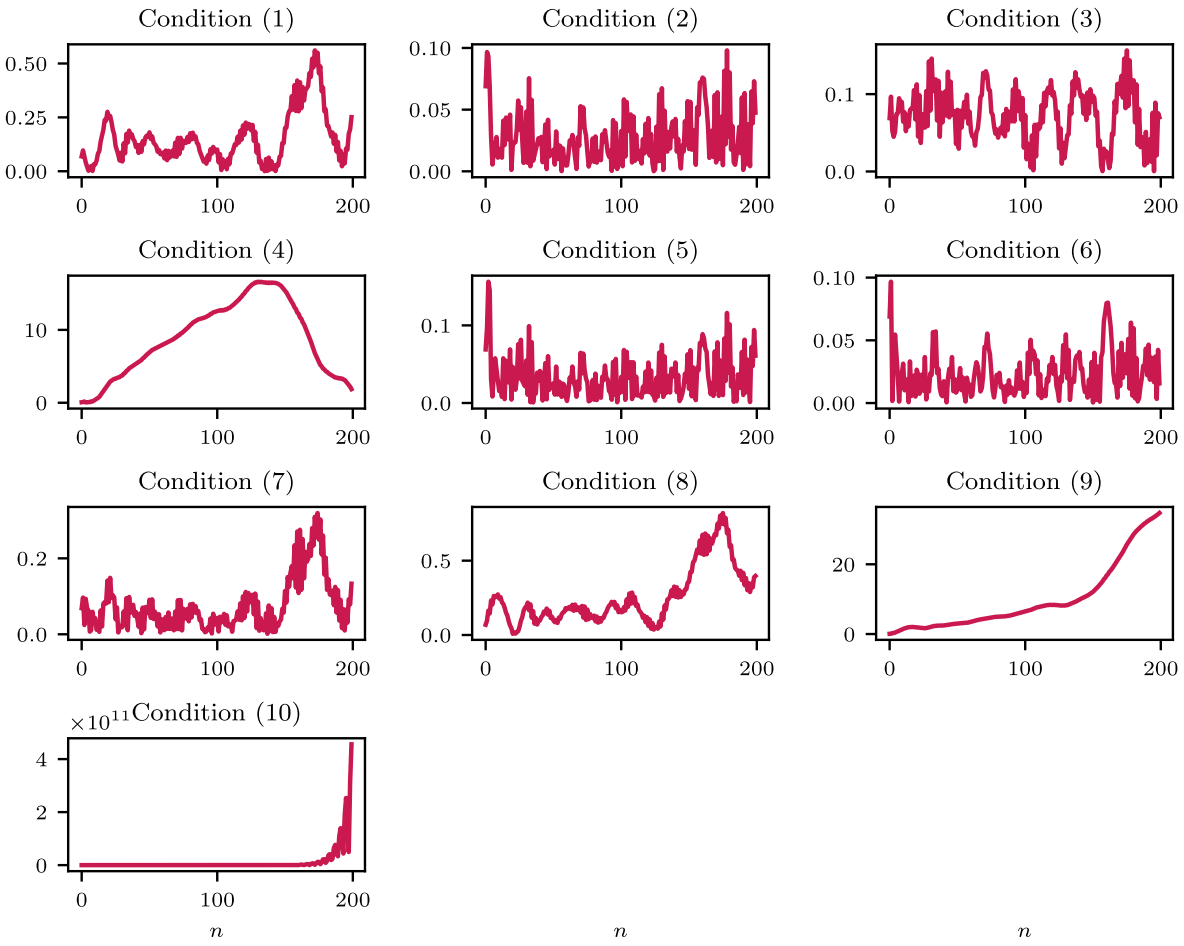


Figure 12.3: Solution at the first point $|u_0^n|$ of the leap-from scheme as function of the time n for the conditions in Table 12.2.

on the right and the boundary condition to study on the left. This is discretized using $N_x = 200$ discrete points and taking $V/\lambda = -1/2$, so that the left boundary is an outflow and the right one is an inflow. The initial data are taken as

$$u_j^0 = \frac{1}{10}\text{random}(-1,1), \qquad u_j^1 = \frac{1}{10}\text{random}(-1,1),$$

where random$(-1,1)$ denotes randomized numbers between -1 and 1. The final time is $T = 1$. The solutions at the final time are given on Figure 12.2. We observe that for the considered conditions, Condition (6) confirms to be stable, whereas wiggles propagating inside the domain from the left boundary appear for Condition (7), since it is mildly unstable. Still, the unstable mode does not grow too much. Finally, for Condition (10), we observe catastrophic instabilities due to the fact that the unstable mode lies outside the unit circle. For the other conditions in Table 12.2, stability or instability is confirmed. We also plot the trend of the solution $|u_0^n|$ at the left boundary as function of the time n, see Figure 12.3, which allows to distinguish between Godunov-Ryabenkii instabilities and only mild growths.

12.2.3.3 Group velocity and reflection coefficient

We now try to study the "mild" instabilities (those for critical $|z| = 1$) more in detail, following the approach by [Trefethen, 1982, Trefethen, 1984, Trefethen, 1996], by introducing the concept of group velocity and reflection coefficient. Observe that the concept of group velocity makes sense as long as the scheme is non-dissipative, which is the case when handling a leap-frog scheme.

Let us look for solutions of the target transport equation of the plane monochromatic wave form

$$u(t,x) = e^{i\omega t} e^{i\xi x}.$$

In this *ansatz*, we observe that $e^{i\omega\Delta t}$ plays the role of z and $e^{i\xi\Delta x}$ the one of κ. Plugging this into the bulk scheme (12.48) yields the so-called dispersion relation of the leap-frog scheme

$$\sin(\omega\Delta t) = -\frac{V}{\lambda}\sin(\xi\Delta x),$$

which is essentially the characteristic equation $\hat{\Phi}(\xi\Delta x, e^{i\omega\Delta t}) = 0$, where $\hat{\Phi}$ is the amplification polynomial of the scheme, *cf.* (7.41). This is an implicit and not necessarily bijective bond giving $\omega = \omega(\xi)$ and allows to define the group velocity

$$C_{\text{group}}(\xi) = -\frac{\mathrm{d}\omega(\xi)}{\mathrm{d}\xi}.$$

Practically, the group velocity is obtained by differentiating the dispersion relation, yielding

$$\Delta t\cos(\omega\Delta t)\mathrm{d}\omega = -\frac{V}{\lambda}\Delta x\cos(\xi\Delta x)\mathrm{d}\xi, \qquad \text{hence} \qquad C_{\text{group}}(\xi) = V\frac{\cos(\xi\Delta x)}{\cos(\omega\Delta t)}.$$

Let us now analyze the case when κ lies on the unit circle: $|\kappa| = 1$. Hence, there exists $\theta \in \mathbb{R}$ such that $\kappa = e^{i\theta}$. Inserting into the characteristic equation (12.50), has the effect of considering the Fourier transform in space, hence gives

$$\left(\mathsf{z} - \frac{1}{\mathsf{z}}\right) = -\frac{2iV}{\lambda}\sin(\theta),$$

which is noting but a second-order equation in z. Under the CFL condition, solving it provides

$$\mathsf{z}(\theta) = \underbrace{-\frac{iV}{\lambda}\sin(\theta)}_{\text{imaginary part}} \pm \underbrace{\sqrt{1 - \frac{V^2}{\lambda^2}\sin^2(\theta)}}_{\text{real part}}.$$

A simple computation reveals that $|\mathsf{z}(\theta)| = 1$, hence also z is on the unit circle, thus can be written as $\mathsf{z}(\theta) = e^{i\pi(\theta)}$ for a function $\pi(\theta) \in \mathbb{R}$. We have that $\theta = \xi\Delta x$ and that $\pi(\theta) = \omega(\theta)\Delta t$, hence we obtain

$$C_{\text{group}} = V\frac{\cos(\theta)}{\cos(\pi(\theta))} = V\frac{\mathrm{Re}(\kappa)}{\mathrm{Re}(\mathsf{z})}.$$

In particular, this group velocity has to be evaluated for the critical (unstable) values of z and κ. For example, we have that

$$\begin{aligned} C_{\text{group}} &= -V > 0, \qquad \text{for} \quad \kappa = \kappa_-, \\ C_{\text{group}} &= V < 0, \qquad \text{for} \quad \kappa = \kappa_+, \end{aligned}$$

just by recalling that $\kappa_-(1) = -1 = -\kappa_+(1)$, $\kappa_-(-1) = 1 = -\kappa_+(-1)$. Hence, the stable mode is right-going whereas that the unstable one is a left-going mode.

Since we are studying instabilities, we have to allow, unlike (12.52), the unstable mode κ_+ to be present in the mix. We thus consider a solution of the form (12.51). [Trefethen, 1982] interprets boundary conditions as a way of imposing a reflection coefficient between left and right-going waves by linking a_+ and a_- in (12.51). Let us give the example for some boundary conditions out of Table 12.2.

- Condition (1). The proof that this condition is unstable is already provided in [Strikwerda, 2004, Gustafsson et al., 1972, Gustafsson et al., 1995]. The unstable mode is on the unit circle and reads $(\mathsf{z}, \kappa_-, \kappa_+) = (-1, 1, 1)$. Inserting the solution (12.51) into the boundary condition gives

$$a_-(1 - \kappa_-(\mathsf{z})) = -a_+(1 - \kappa_+(\mathsf{z})),$$

which gives the reflection coefficient given by the ratio of the amplitude of the reflected wave a_+ and the one of the incident wave a_-:

$$R(z) = -\frac{1-\kappa_-(z)}{1-\kappa_+(z)}.$$

Following [Trefethen, 1982], this needs to be evaluated on the neighborhood of the critical mode $z \to -1$. To do this, the characteristic equation (12.50) of the bulk scheme can be rewritten

$$\kappa(z)^2 + \frac{\lambda(z^2-1)}{Vz}\kappa(z) - 1 = 0,$$

which is quadratic in $\kappa(z)$. In the vicinity of $z \to -1$, it can be easily seen that the explicit expression for the roots is given by

$$\kappa_\pm(z) = \frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} \pm \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right).$$

With this, we obtain the reflection coefficient on the critical mode

$$R_0 = \lim_{z\to-1} R(z) = \lim_{z\to-1} -\frac{1-\frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} - \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right)}{1-\frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} + \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right)} = +\infty,$$

where the limit has been computed using `sympy` under the assumption $V < 0$. The group velocity on the critical state gives $C_{\text{group}}(z=-1,\kappa=1) = -V > 0$.

- Condition (3). One can easily see that the unstable mode is $(z,\kappa_-,\kappa_+) = (1,-1,1)$. The reflection coefficient is given by

$$R(z) = -\frac{1+\kappa_-(z)^2-2z}{1+\kappa_+(z)^2-2z},$$

and has to be considered in the neighborhood of $z \to 1$, giving

$$R_0 = \lim_{z\to1} R(z) = \lim_{z\to1} -\frac{1+\left(\frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} + \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right)\right)^2-2z}{1+\left(\frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} - \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right)\right)^2-2z} = \frac{1+V/\lambda}{1-V/\lambda} > 0,$$

but finite.

- Condition (7). We recall that the unstable mode is $(z,\kappa_-,\kappa_+) = (-1,1,-1)$. The reflection coefficient is given by

$$R(z) = -\frac{z^2+\left(1+\frac{2V}{\lambda}\right)(\kappa_-(z)-1)z-1}{z^2+\left(1+\frac{2V}{\lambda}\right)(\kappa_+(z)-1)z-1},$$

hence we obtain

$$R_0 = \lim_{z\to-1} R(z) = \lim_{z\to-1} -\frac{z^2+\left(1+\frac{2V}{\lambda}\right)\left(\frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} - \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right)-1\right)z-1}{z^2+\left(1+\frac{2V}{\lambda}\right)\left(\frac{1}{2}\left(-\frac{\lambda(z^2-1)}{Vz} + \sqrt{\frac{\lambda^2(z^2-1)^2}{V^2z^2}+4}\right)-1\right)z-1} = +\infty.$$

The aim of [Trefethen, 1984] has been to link the concept of group velocity to the fact that the solution develops L^2 instabilities. This was done to overcome the really difficult notion of GKS stability, which we do not fully provide in this work. To this end [Trefethen, 1984] introduced the following:

Definition 12.2.1

Let the Finite Difference scheme admit a steady state solution (the actual one will be a superposition of such modes)

$$u_j^n = z_0^n \kappa_0^j$$

for $|z_0| \geq 1$. Then

- u is said to be strictly right-going if

$$|z_0| > 1, |\kappa_0| < 1 \qquad \text{or} \qquad |z_0| = |\kappa_0| = 1, C_{\text{group}} > 0$$

- u is said to be strictly left-going if

$$|z_0| > 1, |\kappa_0| > 1 \qquad \text{or} \qquad |z_0| = |\kappa_0| = 1, C_{\text{group}} < 0$$

With Definition 12.2.1 in mind, [Trefethen, 1984] provides a series of results that clarify the differences between GKS stability and L^2 stability.

Theorem 12.2.2

$$\text{GKS stable scheme} \qquad \leftrightarrows \qquad \text{No right-going steady-state exists}$$

We do not give a precise definition of right-going steady state. To fix ideas, it includes strictly right-going steady states and those with $C_{\text{group}} = 0$. This results says that GKS stability has to do with the absence of steady states travelling inside the domain from the boundary or which are stationary at the boundary.

Theorem 12.2.3

GKS stability does not necessarily imply L^2 stability.

We have the following results for zero boundary data, which apply to the schemes we have considered with random initial datum. If the reflection coefficient is bounded, then

Theorem 12.2.4

$$\text{If a strictly right-going steady-state exists} \qquad \rightarrow \qquad \begin{array}{c}\text{the scheme is } L^2 \text{ unstable with growth} \\ \text{at least } \propto \sqrt{n}.\end{array}$$

Instead, if the reflection coefficient if infinite, we have a stronger growth rate.

Theorem 12.2.5

$$\begin{array}{c}\text{If a strictly right-going steady-state} \\ \text{with } R_0 = +\infty \text{ exists}\end{array} \qquad \rightarrow \qquad \begin{array}{c}\text{the scheme is } L^2 \text{ unstable with growth} \\ \text{at least } \propto n.\end{array}$$

With these results in mind, we can revisit the results shown on Figure 12.2. For Condition (1), the unstable growth of the solution is clear, because indeed we have $R_0 = +\infty$, hence at least linear growth in time. For Condition (3), the growth is almost invisible and $R_0 > 0$ but finite, which explains the observation by a rate of growth which should be of order $\sqrt{n}$. The setting of Condition (7) is analogous to that of Condition (1).

12.2.4 Numerical study on the original lattice Boltzmann scheme

For the boundary conditions in Table 12.2 which originate from a boundary condition for the D_1Q_2 scheme for $s = 2$ and which are used to enforce transparent boundary conditions, we repeat the experiment of Section 12.2.3.2 with

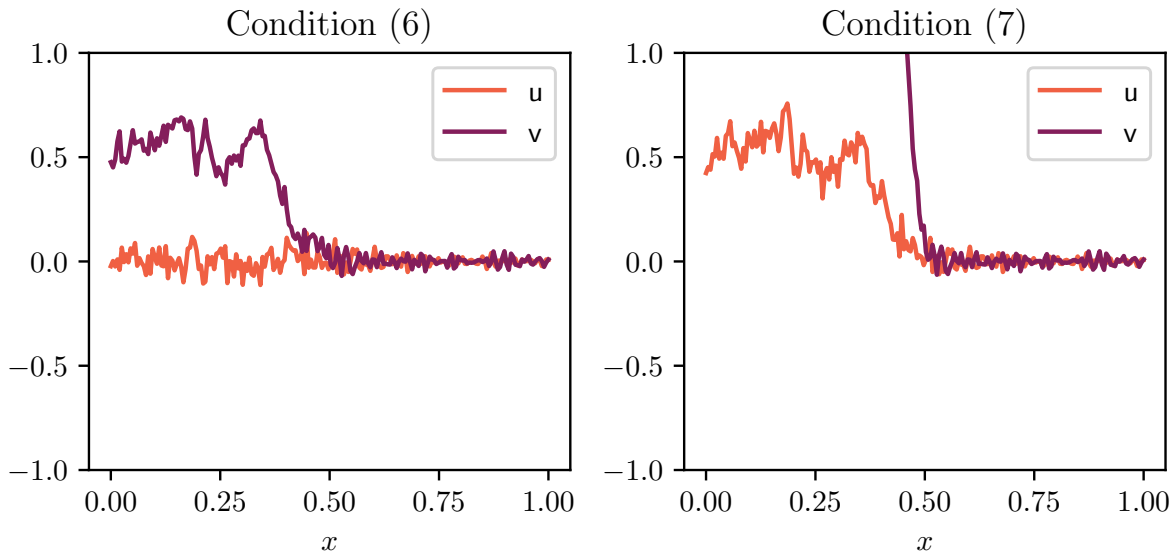


Figure 12.4: Solution of D_1Q_2 scheme both for conserved and non-conserved moment for $s = 2$ at the final time $T = 1$ for two conditions in Table 12.2.

random initial data. These conditions are the 0th order extrapolation Condition (6) and the 1st order extrapolation Condition (7). We take

$$\mathsf{u}_j^0 = \frac{1}{10}\text{random}(-1,1), \qquad \mathsf{v}_j^0 = v^{\text{eq}}(\mathsf{u}_j^0) = V\mathsf{u}_j^0.$$

From the results of Figure 12.4, we observe that even if the have proved that Condition (6) is stable for the conserved moment u, the simulation seems to suggest that it is not stable for the non-conserved moment v, for a spurious mode propagates forward inside the domain at speed 1/2. Quite the opposite, the result for Condition (7) is not surprising. The instabilities on v cannot be predicted using Proposition 12.2.1 and Proposition 12.2.3, since this moment has been eliminated to yield the corresponding Finite Difference scheme. We observe that thanks to the fact that we are in a totally linear setting, if we suppose linearity from the very beginning, the bulk Finite Difference scheme for the non-conserved variable v is again a leap-frog scheme. This provides

$$\left(\mathsf{z} - \frac{1}{\mathsf{z}}\right)\tilde{\mathsf{v}}_j = \frac{V}{\lambda}(\tilde{\mathsf{v}}_{j-1} - \tilde{\mathsf{v}}_{j+1}), \tag{12.55}$$

So the question is: why is the non-conserved moment v different from the conserved moment u taking into account that it satisfies the same discrete scheme? The difference as the simulation goes on will come from the different initial condition and boundary conditions. Still, up to this, we have checked that it is—within machine precision and taking boundary and initial conditions into account—true that v satisfies the same scheme as u. Hence, in order to study the stability of the boundary condition on v, we use the same characteristic equation as for u. We try to rewrite the boundary scheme for the 0th order extrapolation only on v. We were not able to do it in full generality except for the case being the one we tested, that is for $s = 2$, $\lambda = 1$ and $V = -1/2$. One has

$$\mathsf{u}_0^{n+1} = \mathsf{u}_1^n + \frac{1}{2}(\mathsf{v}_1^n - \mathsf{v}_0^n), \tag{12.56}$$

$$\mathsf{v}_0^{n+1} = -\mathsf{u}_1^n - \frac{1}{2}(\mathsf{v}_1^n + \mathsf{v}_0^n), \tag{12.57}$$

$$\mathsf{u}_1^{n+1} = \mathsf{u}_2^n + \frac{1}{2}(\mathsf{v}_2^n - \mathsf{v}_0^n), \tag{12.58}$$

$$\mathsf{v}_1^{n+1} = -\mathsf{u}_2^n - \frac{1}{2}(\mathsf{v}_2^n + \mathsf{v}_0^n). \tag{12.59}$$

Using (12.59) at the previous time, we obtain $\mathsf{u}_2^{n-1} = -\mathsf{v}_1^n - \frac{1}{2}(\mathsf{v}_2^{n-1} + \mathsf{v}_0^{n-1})$, hence plugging into (12.58) at the previous time provides $\mathsf{u}_1^n = -\mathsf{v}_1^n - \mathsf{v}_0^{n-1}$, which put into (12.57) finally yields

$$\mathsf{v}_0^{n+1} = \mathsf{v}_0^{n-1} + \frac{1}{2}(\mathsf{v}_1^n - \mathsf{v}_0^n),$$

which corresponds to nothing but $\mathsf{v}_0^{n+1} = \mathsf{v}_0^{n-1} - \frac{V}{\lambda}(\mathsf{v}_1^n - \mathsf{v}_0^n)$, which is what results on u, see Table 12.2, for the anti-bounce-back condition, which we proved to be unstable when $V < 0$. This explains why we observe the instability on v.

Now the question is: why this instability on the boundary for v does not affect u, since these moments are coupled? The answer lies in the fact that unstable modes are generally "checkerboard" modes of the form

$$\mathsf{v}_j^n \sim (-1)^j,$$

which lie in the unobservable subspace—see Definition 10.4.1 and Example 10.4.1—for this scheme. This explains why the instability cannot be observed on u.

12.3 D_1Q_3 scheme with two conserved moments

We finish by considering a D_1Q_3 scheme (*cf.* Section 1.5.2 with moment matrix $\boldsymbol{M}$ given by (1.5)) with two conserved variables $N = 2$ that we shall indicate u and v, with a third non-conserved moment w. We shall call $s = s_3$. The boundary conditions we enforce are local and come under the form

$$\mathsf{f}_0^{\circ,n+1} = \mathsf{f}_0^{\circ,n,\star}, \tag{12.60}$$

$$\mathsf{f}_0^{+,n+1} = \beta^\circ \mathsf{f}_0^{\circ,n,\star} + \beta^+ \mathsf{f}_0^{+,n,\star} + \beta^- \mathsf{f}_0^{-,n,\star} + \mathsf{S}_0^{n+1}, \tag{12.61}$$

$$\mathsf{f}_0^{-,n+1} = \mathsf{f}_1^{-,n,\star}. \tag{12.62}$$

12.3.1 Continuous problem

Under acoustic scaling and selecting $w^{\mathrm{eq}} = V^2 \mathsf{u}$, the scheme can be used to simulate the wave equation

$$\partial_{tt} u - V^2 \partial_{xx} u = 0, \qquad \text{or equivalently} \qquad \begin{cases} \partial_t u + \partial_x v = 0, \\ \partial_t v + V^2 \partial_x u = 0. \end{cases}$$

Diagonalizing this system of linear conservation laws gives the eigenvalues $\pm V$ with eigenvectors $(\pm 1/V, 1)^{\mathrm{t}}$. Hence the waves of the system satisfy the constraints $Vu \mp v = 0$ and the solution can be decomposed along the eigenvector basis as:

$$\begin{bmatrix} u \\ v \end{bmatrix}(t,x) = \begin{bmatrix} -\frac{1}{V} \\ 1 \end{bmatrix} \phi^-(x + Vt) + \begin{bmatrix} \frac{1}{V} \\ 1 \end{bmatrix} \phi^+(x - Vt),$$

where the scalar function $\phi^\pm$ are determined by the initial datum.

12.3.2 Elimination of the non-conserved moment

With the same techniques as the D_1Q_2, we can eliminate w from the boundary equation to have a Finite Difference scheme on the boundary.

Proposition 12.3.1: Corresponding Finite Difference scheme

Under the condition

$$(\beta^+ + \beta^- - 2\beta^\circ)^2 = 1,$$

the corresponding Finite Difference scheme for the D_1Q_3 scheme endowed with boundary conditions (12.61) reads

$$\begin{aligned}
\mathsf{u}_0^{n+1} &= \Big((1+\beta^\circ) + \frac{(1-s)}{2}(\beta^+ + \beta^- - 2\beta^\circ)\Big)\mathsf{u}_0^n + \frac{(1-s)}{2}\mathsf{u}_1^n - \frac{(1-s)}{2}\Big((\beta^+ + \beta^-)\mathsf{u}_0^{n-1} + \mathsf{u}_1^{n-1}\Big) \\
&+ \frac{1}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^n - \mathsf{v}_1^n\Big) - \frac{(1-s)}{2\lambda}\Big((\beta^+ - \beta^-)\mathsf{v}_0^{n-1} - \mathsf{v}_1^{n-1}\Big) + \frac{s}{2\lambda^2}\Big((\beta^+ + \beta^- - 2\beta^\circ - 2)\mathsf{w}_0^{\mathrm{eq},n} + \mathsf{w}_1^{\mathrm{eq},n}\Big) \\
&+ \mathsf{S}_0^{n+1} - (1-s)\mathsf{S}_0^n.
\end{aligned}$$

$$\begin{aligned}
\mathsf{u}_j^{n+1} &= \mathsf{u}_j^n + \frac{(1-s)}{2}(\mathsf{u}_{j-1}^n + \mathsf{u}_{j+1}^n) - \frac{(1-s)}{2}(\mathsf{u}_{j-1}^{n-1} + \mathsf{u}_{j+1}^{n-1}) + \frac{1}{2\lambda}(\mathsf{v}_{j-1}^n - \mathsf{v}_{j+1}^n) - \frac{(1-s)}{2\lambda}(\mathsf{v}_{j-1}^{n-1} - \mathsf{v}_{j+1}^{n-1}) \\
&\quad + \frac{s}{2\lambda^2}(\mathsf{w}_{j-1}^{\mathrm{eq},n} - 2\mathsf{w}_j^{\mathrm{eq},n} + \mathsf{w}_{j+1}^{\mathrm{eq},n}), \qquad j \geq 1,
\end{aligned}$$

$$\begin{aligned}
\mathsf{v}_0^{n+1} &= \lambda\beta^\circ\Big(1 + (1-s)(\beta^+ + \beta^- - 2\beta^\circ)\Big)\mathsf{u}_0^n + \frac{1}{2}\Big(((\beta^+ - \beta^-) - (1-s)(\beta^+ + \beta^- - 2\beta^\circ))\mathsf{v}_0^n + (2-s)\mathsf{v}_1^n\Big) \\
&\quad + \frac{(1-s)}{2}((\beta^+ - \beta^-)(\beta^+ + \beta^- - 2\beta^\circ) - 1)\mathsf{v}_0^{n-1} + \frac{s}{2\lambda}\Big((\beta^+ + \beta^- - 2\beta^\circ)\mathsf{w}_0^{\mathrm{eq},n} - \mathsf{w}_1^{\mathrm{eq},n}\Big) \\
&\quad + \lambda\mathsf{S}_0^{n+1} + \lambda(1-s)(\beta^+ + \beta^- - 2\beta^\circ)\mathsf{S}_0^n.
\end{aligned}$$

$$\mathsf{v}_j^{n+1} = \frac{(2-s)}{2}(\mathsf{v}_{j-1}^n + \mathsf{v}_{j+1}^n) - (1-s)\mathsf{v}_j^{n-1} + \frac{s}{2\lambda}(\mathsf{w}_{j-1}^{\mathrm{eq},n} - \mathsf{w}_{j+1}^{\mathrm{eq},n}), \qquad j \geq 1.$$

12.3.3 Consistency

We now try to enforce transparent boundary conditions to make waves from inside the domain exiting from the left boundary without reflection. We use Taylor expansions on the corresponding Finite Difference scheme at the boundary. We came to the same conclusions using the Maxwell iteration. The computations are however not presented in this dissertation.

- Consider the oth order extrapolation, which is obtained by $\beta^+ = 1$ and $\beta^- = \beta^\circ = 0$. Applying Taylor expansions to the boundary scheme provides.

$$sS - s\frac{\Delta x}{\lambda}\partial_t u + \frac{\Delta x}{\lambda}\partial_t S + \frac{sV^2}{2\lambda^2}\Delta x\partial_x u - \frac{s}{2\lambda}\Delta x\partial_x v = O(\Delta x^2), \tag{12.63}$$

$$\lambda(2-s)S - \frac{\Delta x}{\lambda}\partial_t v + \Delta x\partial_t S - \frac{sV^2}{2\lambda}\Delta x\partial_x u + \frac{1}{2}(2-s)\Delta x\partial_x v = O(\Delta x^2) \tag{12.64}$$

Taking $S \equiv 0$, we are left with

$$\partial_t u - \frac{V^2}{2\lambda}\partial_x u + \frac{1}{2}\partial_x v = O(\Delta x), \tag{12.65}$$

$$\partial_t v + \frac{sV^2}{2}\partial_x u - \frac{\lambda}{2}(2-s)\partial_x v = O(\Delta x). \tag{12.66}$$

Using the bulk modified equation to eliminate the time derivatives provides

$$\frac{V^2}{\lambda}\partial_x u + \partial_x v = O(\Delta x), \tag{12.67}$$

$$\frac{V^2}{\lambda}(2-s)\partial_x u + (2-s)\partial_x v = O(\Delta x). \tag{12.68}$$

These two equations (up to the formal division by $2-s$) are redundant. Still, we see that this corresponds to

$$\partial_x\Big(\frac{V^2}{\lambda}u + v\Big) = O(\Delta x),$$

which should be compared with the relation for the outgoing wave $Vu + v = 0$. We see that the quantity inside parenthesis does not have the same form, therefore we have to expect that a reflected wave shall form when utilizing this condition. This has already been observed, see [Najafi-Yazdi and Mongeau, 2012].

We might try to solve the issue and obtain transparent boundary conditions by taking $S \not\equiv 0$ with $S = O(\Delta x)$. In particular, consider $S = \Delta x \alpha \partial_x u$, where $\partial_x u$ shall be an approximation of the derivative of the first conserved moment at the boundary. We have to select the parameter α. We have

$$s\Delta x\alpha\partial_x u - s\frac{\Delta x}{\lambda}\partial_t u + \frac{sV^2}{2\lambda^2}\Delta x\partial_x u - \frac{s}{2\lambda}\Delta x\partial_x v = O(\Delta x^2), \tag{12.69}$$

$$\lambda(2-s)\Delta x\alpha\partial_x u - \frac{\Delta x}{\lambda}\partial_t v - \frac{sV^2}{2\lambda}\Delta x\partial_x u + \frac{1}{2}(2-s)\Delta x\partial_x v = O(\Delta x^2) \tag{12.70}$$

Again using the bulk equation to get rid of the time derivatives

$$2\lambda\alpha\partial_x u + \frac{V^2}{\lambda}\partial_x u + \partial_x v = O(\Delta x), \tag{12.71}$$

$$2\lambda(2-s)\alpha\partial_x u + \frac{1}{\lambda}V^2(2-s)\partial_x u + (2-s)\partial_x v = O(\Delta x). \tag{12.72}$$

As before the relations are redundant. We recast as

$$\partial_x\Big(\Big(2\lambda\alpha + \frac{V^2}{\lambda}\Big)u + v\Big) = O(\Delta x),$$

thus enforcing the identity for the outgoing wave in order to have only it at the boundary: $2\lambda\alpha + V^2/\lambda = V$. Thus the value for α is

$$\alpha = \frac{1}{2}\frac{V}{\lambda}\Big(1 - \frac{V}{\lambda}\Big).$$

Using $\partial_x u \simeq (u_1 - u_0)/\Delta x$ as approximation of the derivative in S, the results that we obtain are of pretty good quality to enforce transparent boundary conditions, see Section 12.3.4. Strictly speaking, this condition is not local because we need to approximate the derivative in order to construct the boundary source term.

Observe that another possibility would have been to consider S proportional to $\partial_x v$ and to proceed in the analogous way. All the combinations of these ways of proceeding are equally interesting.

- Consider the anti-bounce-back condition with $\beta^- = -1$ and $\beta^+ = \beta^\circ = 0$. A Taylor expansion at the boundary gives two strictly identical equations

$$-\frac{V^2}{\lambda^2}u + S = O(\Delta x). \tag{12.73}$$

To enforce a transparent boundary condition by obtaining the relation of the outgoing wave. Let $S = \alpha v$, giving

$$Vu - \frac{\lambda^2\alpha}{V}v = O(\Delta x), \tag{12.74}$$

hence we request $-\lambda^2\alpha/V = 1$, therefore $\alpha = -V/\lambda^2$. Again as the previous one, this solution works pretty well in having a transparent boundary condition.

- Consider the oth order anti extrapolation with $\beta^+ = -1$ and $\beta^- = \beta^\circ = 0$. We have once more two redundant equations at the boundary which read

$$\frac{V^2}{\lambda}u + v - \lambda S = O(\Delta x).$$

We look for $S = \alpha u$. Hence

$$\Big(\frac{V^2}{\lambda} - \lambda\alpha\Big)u + v = O(\Delta x).$$

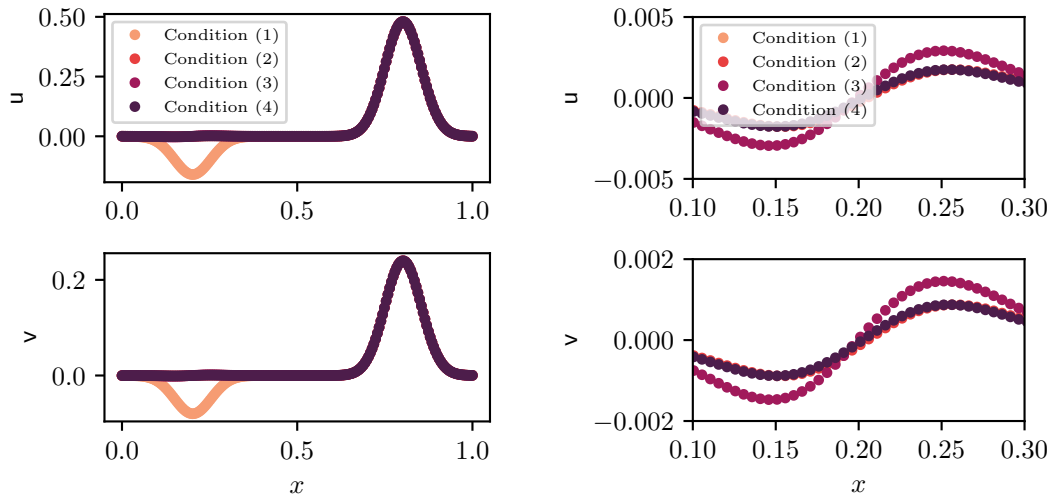


Figure 12.5: Solution of D_1Q_3 at iteration $n = 200$ for different boundary conditions. Magnification on the right.

Therefore we want to consider $\alpha = (V/\lambda)(V/\lambda - 1)$. Again this solution gives satisfying results for the transparent condition.

- Consider the bounce back condition $\beta^- = 1$ and $\beta^+ = \beta^\circ = 0$. The Taylor expansions at the boundary are

$$\frac{s}{\lambda} v - sS = O(\Delta x)$$
$$(2-s)v - \lambda(2-s)S = O(\Delta x)$$

This results in a Dirichlet boundary condition on v. This can be used to enforce the presence of a solid wall or an oscillating wall, see [LeVeque, 2002, Chapter 7]. Still, it can also be used in combination with a source term to enforce absorbing conditions. Overall, all the previous conditions could be used, by working on the source term, to enforce solid or oscillating walls as well.

12.3.4 Numerical tests

We now test some of the transparent conditions that we have proposed in Section 12.3.3. They are:

$$\begin{aligned}
&\text{Condition (1)} && \beta^+ = 1, \beta^- = \beta^\circ = 0, && \text{with} && \mathsf{S}_0^{n+1} = 0,\\
&\text{Condition (2)} && \beta^+ = 1, \beta^- = \beta^\circ = 0, && \text{with} && \mathsf{S}_0^{n+1} = \frac{V}{2\lambda}\Big(1 - \frac{V}{\lambda}\Big)(\mathsf{u}_1^n - \mathsf{u}_0^n),\\
&\text{Condition (3)} && \beta^- = -1, \beta^+ = \beta^\circ = 0, && \text{with} && \mathsf{S}_0^{n+1} = -\frac{V}{\lambda^2}\mathsf{v}_0^n,\\
&\text{Condition (4)} && \beta^+ = -1, \beta^- = \beta^\circ = 0, && \text{with} && \mathsf{S}_0^{n+1} = \frac{V}{\lambda}\Big(\frac{V}{\lambda} - 1\Big)\mathsf{u}_0^n.
\end{aligned}$$

We simulate on the bounded domain $\Omega = [0,1]$ discretized with $N_x = 200$ points. We consider $\lambda = 1$ and $V = 1/2$. The initial datum is a point-wise discretization of

$$u^\circ(x) = \exp\Big(-\frac{(x-0.3)^2}{0.005}\Big), \qquad v^\circ(x) = 0.$$

The results given in Figure 12.5 confirm that Condition (1) produces a large spurious reflected wave because it does not yield the characteristic relation of the exiting wave. Conditions (2), (3) and (4) give very good results with very small reflected waves. Indeed, Condition (2) and (4) give very similar amplitudes, whereas Condition (3) gives slightly larger reflected waves.

12.4 Conclusions and open issues

In this Chapter 12, we have have investigated the consistency and the stability—using a well-known approach for Finite Difference—for very simple boundary conditions and lattice Boltzmann schemes by relying on the corresponding Finite Difference scheme. We have also introduced a formal technique—derived from the Maxwell iteration—to analyze the consistency without having to eliminate the non-conserved moments. This approach yields results compatible with the one on the corresponding Finite Difference scheme, thus confirming that it deserves to be considered, especially to deal with more complex schemes. However, contrarily to the bulk scheme, there is no systematic way of eliminating the non-conserved moments from the scheme, which limits the generality of our studies based on the corresponding Finite Difference scheme.

In terms of future investigations, as far as the consistency is concerned, the Maxwell iteration can be used but one has to keep in mind that it is merely formal. Concerning stability, an extremely important question concerns the way of studying the stability without having a general result which allows to eliminate the non-conserved moments. The first trials employing the GSK theory on the original lattice Boltzmann scheme seen as a one-step scheme for several moments have been—for the moment—unsuccessful.

Conclusions and perspectives of the thesis

This thesis has answered two important questions concerning lattice Boltzmann schemes, which were quite different in their nature. The first one focused on a more practical aspect, namely the efficiency of the numerical simulations and—in particular—the important demand in terms of storage, which can become a stumbling block for real-life applications. The second one was more theoretical, linked to the convergence of general lattice Boltzmann schemes and the possibility of providing a comprehensive framework to study them from the standpoint of numerical analysis. Considering the impressive number of applications of these numerical methods, we believe that the contributions which we have presented in this manuscript will be valuable to a large panel of researchers and practical situations. The work has been articulated around three main axes.

1. Provide a way of using general lattice Boltzmann schemes over dynamically evolving meshes, still ensuring error control. This has been achieved by using adaptive multiresolution in a holistic fashion, that is, both for the grid adaptation and to transform the original lattice Boltzmann scheme in order to use it on these meshes. While this has secured error estimates and evident gains in terms of storage, the proposed strategy has also demonstrated to be extremely reliable in terms of numerical properties, overcoming difficulties which were well-known in the lattice Boltzmann community and over-performing compared to the available approaches. This has important implications since our approach can be employed irrespective of the lattice Boltzmann scheme at hand, ensuring a quantitative control on the quality of simulations, along with a significant reduction of the memory occupation and parasite phenomena due to grid adaptation.
2. Implement the previous approach in a more general C++ library allowing to deal with adaptive meshes—both made up of volumes and points, adapted with AMR, multiresolution, *etc.*—and several classes of numerical schemes. This has led to the development of SAMURAI, which is now used, beyond adaptive lattice Boltzmann schemes, by several researchers to handle Finite Volume schemes for plasma physics, implicit solvers for the Navier-Stokes equations, *etc.* This has been made possible by storing Cartesian meshes in a compressed level-wise fashion using intervals of relative integers and by the implementation of suitable operators on sets acting on this representation.
3. Provide theoretical foundations for lattice Boltzmann schemes based on numerical analysis. In particular, we have focused on formulating exploitable notations of consistency and stability paving the way to a general convergence theory for these schemes. This has been fostered by the elimination of the non-conserved variables from the formulation to yield multi-step Finite Difference schemes. From this, consistency and stability, hence convergence, are inherited from the well-known theory of Finite Difference schemes and can be studied agnostically of the underlying lattice Boltzmann method. Moreover, the role of initialization on numerical simulations has been elucidated using tools germane to numerical analysis. These points constitute a breakthrough in the understanding of lattice Boltzmann schemes, which therefore enter in the framework of "standard" numerical schemes for PDEs such as Finite Difference, Finite Volume and Finite Elements methods. Moreover, the results we have found agree with those already existing in the literature, thus they comfort the frequently heuristic arguments resorting to more mathematically sound justifications.

Still following the trifold articulation of the work, current work in progress and short-term future investigations as well as broader long-term questions are as follows.

1. Use our adaptive lattice Boltzmann method based on multiresolution—which has been thoroughly investigated from the theoretical standpoint—to address applications which naturally call for mesh adaptation. These include multiphase flows, *cf.* Figure 12.6, where one can hope to finely mesh exclusively close to the

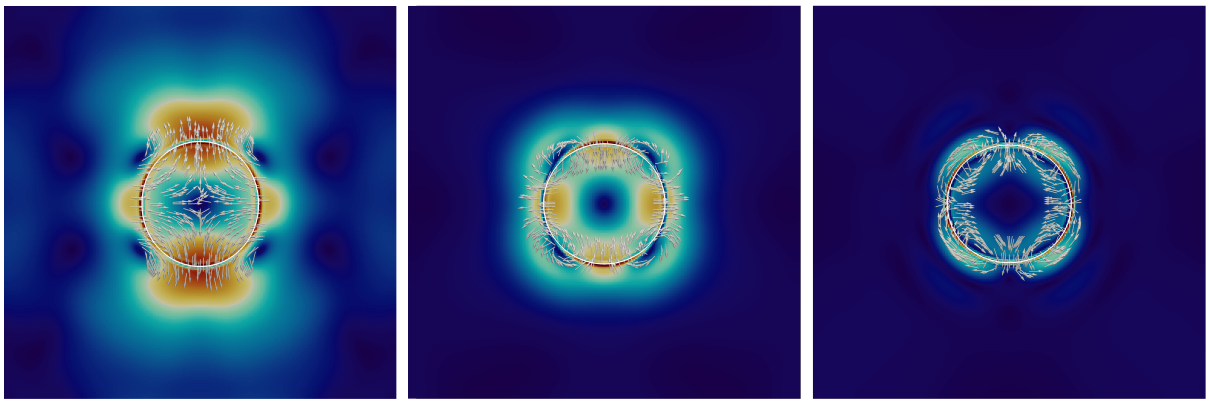


Figure 12.6: Simulation of an oscillating bubble on a uniform mesh using a phase-field lattice Boltzmann method proposed by [Fakhari et al., 2016]. The interface is indicated by a white line. Colors indicate the norm of the velocity field. Arrows point as the velocity field.

interface without sacrificing accuracy, and compressible flows (*i.e.* modeled using the compressible Navier-Stokes equations), especially in the hypersonic regime, where concentrated steep transitions in the solutions are expected to appear.

2. The work done on SAMURAI can be enhanced by dealing with the parallelization of the currently sequential code, in order to fully exploit the capabilities of modern computer architectures. Another path of improvement concerns the way mesh adaptation using multiresolution is implemented, which can be optimized to reduce the time spent on this operation at each iteration of the numerical schemes. Moreover, the current implementation of adaptive lattice Boltzmann schemes offers room for improvement, for it relies on certain choices that have been previously described and that can be revalued to gain additional computational efficiency. Furthermore, considering that GPUs are nowadays the state-of-the-art architecture for deploying lattice Boltzmann schemes, the implementation of SAMURAI and the related code on GPUs will be also investigated. From a broader perspective, we will pursue interactions with the other members of our team in order to shape SAMURAI to handle linear systems stemming from the Poisson problem (plasma physics), treat problems regarding two-phase flows, solve the Navier-Stokes equations with more traditional numerical methods, *etc.*

3. Finally, as far as the theoretical investigation of lattice Boltzmann schemes is concerned, we still face many open and exciting challenges. Future paths for research—that have been initiated in this thesis—include rigorous analyses of the stability of lattice Boltzmann schemes with respect to other norms than the L^2. We hope that the corresponding multi-step Finite Difference schemes be helpful on this matter, even if less theory on these schemes is available, as far as these norms are concerned. Another interesting but extremely vast point concerns boundary conditions. We have seen that our approach, relying on corresponding Finite Difference schemes, does not seem to be very helpful, at least beyond simple schemes. We do believe that scientists ought to propose paradigm-shifting tools to deal with this issue. Finally, as mesoscopic systems can exhibit interesting behavior (non-conservative products appearing in the macroscopic equations, *etc.*) [Graille et al., 2009] due to the presence of multiple scales (mass-ratios and time-ratios) in the problem, and lattice Boltzmann schemes can be interpreted as discretizations of finite-velocities Boltzmann equations, an interesting topic will be to study what multi-scale lattice Boltzmann schemes would look like and what their properties would be.

Bibliography

[Abramowitz and Stegun, 1964] Abramowitz, M. and Stegun, I. A. (1964). *Handbook of mathematical functions with formulas, graphs, and mathematical tables*. Dover Books.

[Alauzet et al., 2003] Alauzet, F., George, P. L., Mohammadi, B., Frey, P., and Borouchaki, H. (2003). Transient fixed point-based unstructured mesh adaptation. *International Journal for Numerical Methods in Fluids*, 43(6-7):729–745.

[Allaire, 2007] Allaire, G. (2007). *Numerical analysis and optimization: an introduction to mathematical modelling and numerical simulation*. Oxford University Press.

[Ambrosio et al., 2000] Ambrosio, L., Fusco, N., and Pallara, D. (2000). *Functions of bounded variation and free discontinuity problems*. Oxford Science Publications.

[Aregba-Driollet and Natalini, 2000] Aregba-Driollet, D. and Natalini, R. (2000). Discrete kinetic schemes for multidimensional systems of conservation laws. *SIAM Journal on Numerical Analysis*, 37(6):1973–2004.

[Astoul et al., 2021] Astoul, T., Wissocq, G., Boussuge, J.-F., Sengissen, A., and Sagaut, P. (2021). Lattice Boltzmann method for computational aeroacoustics on non-uniform meshes: A direct grid coupling approach. *Journal of Computational Physics*, 447:110667.

[Åström and Murray, 2008] Åström, K. J. and Murray, R. M. (2008). *Feedback systems: an introduction for scientists and engineers*. Princeton University Press.

[Axner et al., 2008] Axner, L., Bernsdorf, J., Zeiser, T., Lammers, P., Linxweiler, J., and Hoekstra, A. G. (2008). Performance evaluation of a parallel sparse lattice Boltzmann solver. *Journal of Computational Physics*, 227(10):4895–4911.

[Banda et al., 2006] Banda, M. K., Yong, W.-A., and Klar, A. (2006). A stability notion for lattice Boltzmann equations. *SIAM Journal on Scientific Computing*, 27(6):2098–2111.

[Barsukow and Abgrall, 2023] Barsukow, W. and Abgrall, R. (2023). Extensions of Active Flux to arbitrary order of accuracy. *ESAIM: Mathematical Modelling and Numerical Analysis*, 57(2):991–1027.

[Baty et al., 2023] Baty, H., Drui, F., Helluy, P., Franck, E., Klingenberg, C., and Thanhäuser, L. (2023). A robust and efficient solver based on kinetic schemes for Magnetohydrodynamics (MHD) equations. *Applied Mathematics and Computation*, 440:127667.

[Bellotti, 2023a] Bellotti, T. (2023a). Initialisation from lattice Boltzmann to multi-step Finite Difference methods: modified equations and discrete observability. *Submitted, see https://hal.science/hal-03989355*.

[Bellotti, 2023b] Bellotti, T. (2023b). Truncation errors and modified equations for the lattice Boltzmann method via the corresponding Finite Difference schemes. *ESAIM: Mathematical Modelling and Numerical Analysis*, 57(3):1225–1255.

[Bellotti et al., 2022a] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022a). Does the multiresolution lattice Boltzmann method allow to deal with waves passing through mesh jumps? *Comptes Rendus. Mathématique*, 360:761–769.

[Bellotti et al., 2022b] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022b). High accuracy analysis of adaptive multiresolution-based lattice Boltzmann schemes via the equivalent equations. *SMAI Journal of Computational Mathematics*, 8:161–199.

[Bellotti et al., 2022c] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022c). Multidimensional fully adaptive lattice Boltzmann methods with error control based on multiresolution analysis. *Journal of Computational Physics*, 471:111670.

[Bellotti et al., 2022d] Bellotti, T., Gouarin, L., Graille, B., and Massot, M. (2022d). Multiresolution-based mesh adaptation and error control for lattice Boltzmann methods with applications to hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 44(4):A2599–A2627.

[Bellotti et al., 2023a] Bellotti, T., Gouarin, L., Leclerc, H., Massot, M., and Séries, L. (2023a). Interval-based data structure for Cartesian meshes: application to multi-scale PDEs on adaptive meshes. *In Preparation*.

[Bellotti et al., 2022e] Bellotti, T., Graille, B., and Massot, M. (2022e). Finite difference formulation of any lattice Boltzmann scheme. *Numerische Mathematik*, 152:1–40.

[Bellotti et al., 2023b] Bellotti, T., Massot, J., Massot, M., Séries, L., and Tenaud, C. (2023b). Modified equation and error analyses of adaptive multiresolution Finite Volume schemes. *In Preparation*.

[Bender et al., 1999] Bender, C. M., Orszag, S., and Orszag, S. A. (1999). *Advanced mathematical methods for scientists and engineers I: Asymptotic methods and perturbation theory*, volume 1. Springer Science & Business Media.

[Benzi et al., 1992] Benzi, R., Succi, S., and Vergassola, M. (1992). The lattice Boltzmann equation: theory and applications. *Physics Reports*, 222(3):145–197.

[Berger and Oliger, 1984] Berger, M. J. and Oliger, J. (1984). Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53(3):484–512.

[Bergounioux, 2011] Bergounioux, M. (2011). On Poincaré-Wirtinger inequalities in spaces of functions of bounded variation. *Control and Cybernetics*, 40(4):921–930.

[Bernaschi et al., 2010] Bernaschi, M., Fatica, M., Melchionna, S., Succi, S., and Kaxiras, E. (2010). A flexible high-performance Lattice Boltzmann GPU code for the simulations of fluid flows in complex geometries. *Concurrency and Computation: Practice and Experience*, 22(1):1–14.

[Bihari and Harten, 1997] Bihari, B. L. and Harten, A. (1997). Multiresolution schemes for the numerical solution of 2-D conservation laws I. *SIAM Journal on Scientific Computing*, 18(2):315–354.

[Blyth, 2018] Blyth, T. S. (2018). *Module theory: an approach to linear algebra*. University of St Andrews.

[Boghosian et al., 2018] Boghosian, B., Dubois, F., Graille, B., Lallemand, P., and Tekitek, M.-M. (2018). Curious Convergence Properties of Lattice Boltzmann Schemes for Diffusion with Acoustic Scaling. *Communications in Computational Physics*, 23(4):1263–1278.

[Bouchut, 2004] Bouchut, F. (2004). *Nonlinear stability of finite Volume Methods for hyperbolic conservation laws: And Well-Balanced schemes for sources*. Springer Science & Business Media.

[Bouchut et al., 2000] Bouchut, F., Guarguaglini, F. R., and Natalini, R. (2000). Diffusive BGK approximations for nonlinear multidimensional parabolic equations. *Indiana University Mathematics Journal*, pages 723–749.

[Bouzidi et al., 2001] Bouzidi, M., Firdaouss, M., and Lallemand, P. (2001). Momentum transfer of a Boltzmann-lattice fluid with boundaries. *Physics of Fluids*, 13(11):3452–3459.

[Bramkamp et al., 2004] Bramkamp, F., Lamby, P., and Müller, S. (2004). An adaptive multiscale finite volume solver for unsteady and steady state flow computations. *Journal of Computational Physics*, 197(2):460–490.

[Brenier, 1984] Brenier, Y. (1984). Averaged multivalued solutions for scalar conservation laws. *SIAM Journal on Numerical Analysis*, 21(6):1013–1037.

[Brenner et al., 1975] Brenner, P., Thomée, V., and Wahlbin, L. B. (1975). *Besov Spaces and Applications to Difference Methods for Initial Value Problems*. Springer.

[Brewer et al., 1986] Brewer, J. W., Bunce, J. W., and Van Vleck, F. S. (1986). *Linear systems over commutative rings*. CRC Press.

[Brézis, 2011] Brézis, H. (2011). *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer.

[Brix et al., 2011] Brix, K., Melian, S., Müller, S., and Bachmann, M. (2011). Adaptive multiresolution methods: Practical issues on data structures, implementation and parallelization. In *ESAIM: Proceedings*, volume 34, pages 151–183. EDP Sciences.

[Broadwell, 1964] Broadwell, J. E. (1964). Study of rarefied shear flow by the discrete velocity method. *Journal of Fluid Mechanics*, 19(3):401–414.

[Bürger et al., 2008] Bürger, R., Ruiz, R., Schneider, K., and Sepúlveda, M. A. (2008). Fully adaptive multiresolution schemes for strongly degenerate parabolic equations with discontinuous flux. *Journal of Engineering Mathematics*, 60(3):365–385.

[Burstedde et al., 2011] Burstedde, C., Wilcox, L. C., and Ghattas, O. (2011). p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM Journal on Scientific Computing*, 33(3):1103–1133.

[Cabannes et al., 1980] Cabannes, H., Gatignol, R., and Luol, L. (1980). The discrete Boltzmann equation. *Lecture Notes at University of California, Berkley*, pages 1–65.

[Caetano et al., 2023] Caetano, F., Dubois, F., and Graille, B. (2023). A result of convergence for a mono-dimensional two-velocities lattice Boltzmann scheme. *Accepted in Discrete and Continuous Dynamical Systems Series S*.

[Caiazzo, 2005] Caiazzo, A. (2005). Analysis of lattice Boltzmann initialization routines. *Journal of Statistical Physics*, 121(1):37–48.

[Caiazzo et al., 2009] Caiazzo, A., Junk, M., and Rheinländer, M. (2009). Comparison of analysis techniques for the lattice Boltzmann method. *Computers & Mathematics with Applications*, 58(5):883–897.

[Carpentier et al., 1997] Carpentier, R., de La Bourdonnaye, A., and Larrouturou, B. (1997). On the derivation of the modified equation for the analysis of linear numerical methods. *ESAIM: Mathematical Modelling and Numerical Analysis*, 31(4):459–470.

[Chambolle and Pock, 2021] Chambolle, A. and Pock, T. (2021). Learning consistent discretizations of the total variation. *SIAM Journal on Imaging Sciences*, 14(2):778–813.

[Chapman and Cowling, 1990] Chapman, S. and Cowling, T. G. (1990). *The mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction and diffusion in gases*. Cambridge University Press.

[Chen, 1998] Chen, H. (1998). Volumetric formulation of the lattice Boltzmann method for fluid dynamics: Basic concept. *Physical Review E*, 58(3):3955.

[Chen et al., 1991] Chen, S., Chen, H., Martnez, D., and Matthaeus, W. (1991). Lattice Boltzmann model for simulation of magnetohydrodynamics. *Physical Review Letters*, 67(27):3776.

[Chen and Doolen, 1998] Chen, S. and Doolen, G. D. (1998). Lattice Boltzmann method for fluid flows. *Annual Review of Fluid Mechanics*, 30(1):329–364.

[Cheng, 2003] Cheng, S. S. (2003). *Partial difference equations*, volume 3. CRC Press.

[Cheng and Lu, 1999] Cheng, S. S. and Lu, Y.-F. (1999). General solutions of a three-level partial difference equation. *Computers & Mathematics with Applications*, 38(7-8):65–79.

[Chiavassa and Donat, 2001] Chiavassa, G. and Donat, R. (2001). Point value multiscale algorithms for 2D compressible flows. *SIAM Journal on Scientific Computing*, 23(3):805–823.

[Cohen et al., 2004] Cohen, A., Dahmen, W., and DeVore, R. (2004). Adaptive wavelet techniques in numerical simulation. *Encyclopedia of Computational Mechanics*, 1:157–197.

[Cohen et al., 1992] Cohen, A., Daubechies, I., and Feauveau, J.-C. (1992). Biorthogonal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 45(5):485–560.

[Cohen et al., 2003] Cohen, A., Kaber, S., Müller, S., and Postel, M. (2003). Fully adaptive multiresolution finite volume schemes for conservation laws. *Mathematics of Computation*, 72(241):183–225.

[Cole, 1951] Cole, J. D. (1951). On a quasi-linear parabolic equation occurring in aerodynamics. *Quarterly of Applied Mathematics*, 9(3):225–236.

[Colella, 1990] Colella, P. (1990). Multidimensional upwind methods for hyperbolic conservation laws. *Journal of Computational Physics*, 87(1):171–200.

[Coquel et al., 2010] Coquel, F., Nguyen, Q. L., Postel, M., and Tran, Q. H. (2010). Local time stepping applied to implicit-explicit methods for hyperbolic systems. *Multiscale Modeling & Simulation*, 8(2):540–570.

[Coquel et al., 2006] Coquel, F., Postel, M., Poussineau, N., and Tran, Q. H. (2006). Multiresolution technique and explicit–implicit scheme for multicomponent flows. *Journal of Numerical Mathematics*, 14(3):187–216.

[Coreixas et al., 2019] Coreixas, C., Chopard, B., and Latt, J. (2019). Comprehensive comparison of collision models in the lattice Boltzmann framework: Theoretical investigations. *Physical Review E*, 100(3):033305.

[Coulombel, 2009] Coulombel, J.-F. (2009). Stability of finite difference schemes for hyperbolic initial boundary value problems. *SIAM Journal on Numerical Analysis*, 47(4):2844–2871.

[Coulombel, 2011a] Coulombel, J.-F. (2011a). Stability of finite difference schemes for hyperbolic initial boundary value problems. Lecture notes.

[Coulombel, 2011b] Coulombel, J.-F. (2011b). Stability of finite difference schemes for hyperbolic initial boundary value problems II. *Annali della Scuola Normale Superiore di Pisa-Classe di Scienze*, 10(1):37–98.

[Courtès, 2017] Courtès, C. (2017). *Analyse numérique de systèmes hyperboliques-dispersifs*. PhD thesis, Université Paris-Saclay.

[Crouse et al., 2003] Crouse, B., Rank, E., Krafczyk, M., and Tölke, J. (2003). A LB-based approach for adaptive flow simulations. *International Journal of Modern Physics B*, 17(01n02):109–112.

[Cull et al., 2005] Cull, P., Flahive, M., and Robson, R. (2005). Matrix Difference Equations. *Difference Equations: From Rabbits to Chaos*, pages 179–216.

[Daru and Tenaud, 2004] Daru, V. and Tenaud, C. (2004). High order one-step monotonicity-preserving schemes for unsteady compressible flow calculations. *Journal of Computational Physics*, 193(2):563–594.

[Daubechies, 1988] Daubechies, I. (1988). Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7):909–996.

[Daubechies, 1992] Daubechies, I. (1992). *Ten lectures on wavelets*. SIAM.

[Davis, 2006] Davis, T. A. (2006). *Direct methods for sparse linear systems*. SIAM.

[De Schutter, 2000] De Schutter, B. (2000). Minimal state-space realization in linear system theory: an overview. *Journal of Computational and Applied Mathematics*, 121(1-2):331–354.

[Deiterding et al., 2016] Deiterding, R., Domingues, M. O., Gomes, S. M., and Schneider, K. (2016). Comparison of adaptive multiresolution and adaptive mesh refinement applied to simulations of the compressible Euler equations. *SIAM Journal on Scientific Computing*, 38(5):S173–S193.

[Dellacherie, 2014] Dellacherie, S. (2014). Construction and analysis of lattice Boltzmann methods applied to a 1D convection-diffusion equation. *Acta Applicandae Mathematicae*, 131(1):69–140.

[Dellar, 2002] Dellar, P. J. (2002). Lattice kinetic schemes for magnetohydrodynamics. *Journal of Computational Physics*, 179(1):95–126.

[Dellar, 2003] Dellar, P. J. (2003). Incompressible limits of lattice boltzmann equations using multiple relaxation times. *Journal of Computational Physics*, 190(2):351–370.

[Dellar, 2013a] Dellar, P. J. (2013a). An interpretation and derivation of the lattice Boltzmann method using Strang splitting. *Computers & Mathematics with Applications*, 65(2):129–141.

[Dellar, 2013b] Dellar, P. J. (2013b). Lattice Boltzmann magnetohydrodynamics with current-dependent resistivity. *Journal of Computational Physics*, 237:115–131.

[Descombes et al., 2017] Descombes, S., Duarte, M., Dumont, T., Guillet, T., Louvet, V., and Massot, M. (2017). Task-based adaptive multiresolution for time-space multi-scale reaction-diffusion systems on multi-core architectures. *The SMAI Journal of Computational Mathematics*, 3:29–51.

[Descombes et al., 2014] Descombes, S., Duarte, M., Dumont, T., Laurent, F., Louvet, V., and Massot, M. (2014). Analysis of operator splitting in the nonasymptotic regime for nonlinear reaction-diffusion equations. Application to the dynamics of premixed flames. *SIAM Journal on Numerical Analysis*, 52(3):1311–1334.

[DeVore and Sharpley, 1984] DeVore, R. A. and Sharpley, R. C. (1984). *Maximal functions measuring smoothness*, volume 293. American Mathematical Soc.

[D'Humières, 1992] D'Humières, D. (1992). *Generalized Lattice-Boltzmann Equations*, pages 450–458. American Institute of Aeronautics and Astronautics, Inc.

[d'Humières and Ginzburg, 2009] d'Humières, D. and Ginzburg, I. (2009). Viscosity independent numerical errors for Lattice Boltzmann models: From recurrence equations to "magic" collision numbers. *Computers & Mathematics with Applications*, 58(5):823–840.

[Ding and Zhou, 2007] Ding, J. and Zhou, A. (2007). Eigenvalues of rank-one updated matrices with some applications. *Applied Mathematics Letters*, 20(12):1223–1226.

[Donoho, 1992] Donoho, D. L. (1992). Interpolating wavelet transforms. *Preprint, Department of Statistics, Stanford University*, 2(3):1–54.

[Drui, 2017] Drui, F. (2017). *Modélisation et simulation Eulériennes des écoulements diphasiques à phases séparées et dispersées: développement d'une modélisation unifiée et de méthodes numériques adaptées au calcul massivement parallèle.* PhD thesis, Université Paris-Saclay.

[Duarte et al., 2015] Duarte, M., Bonaventura, Z., Massot, M., and Bourdon, A. (2015). A numerical strategy to discretize and solve the Poisson equation on dynamically adapted multiresolution grids for time-dependent streamer discharge simulations. *Journal of Computational Physics*, 289:129–148.

[Duarte et al., 2013] Duarte, M., Descombes, S., Tenaud, C., Candel, S., and Massot, M. (2013). Time–space adaptive numerical methods for the simulation of combustion fronts. *Combustion and Flame*, 160(6):1083–1101.

[Duarte et al., 2012] Duarte, M., Massot, M., Descombes, S., Tenaud, C., Dumont, T., Louvet, V., and Laurent, F. (2012). New resolution strategy for multiscale reaction waves using time operator splitting, space adaptive multiresolution, and dedicated high order implicit/explicit time integrators. *SIAM Journal on Scientific Computing*, 34(1):A76–A104.

[Duarte, 2011] Duarte, M. P. (2011). *Adaptive numerical methods in time and space for the simulation of multi-scale reaction fronts.* PhD thesis, Ecole Centrale Paris.

[Dubois, 2008] Dubois, F. (2008). Equivalent partial differential equations of a lattice Boltzmann scheme. *Computers & Mathematics with Applications*, 55(7):1441–1449.

[Dubois, 2014] Dubois, F. (2014). Simulation of strong nonlinear waves with vectorial lattice Boltzmann schemes. *International Journal of Modern Physics C*, 25(12):1441014.

[Dubois, 2019] Dubois, F. (2019). General third order Chapman-Enskog expansion of lattice Boltzmann schemes. In *16th International Conference for Mesoscopic Methods in Engineering and Science, Edinburgh, 22–26 July 2019.*, Edimburgh, United Kingdom.

[Dubois, 2022] Dubois, F. (2022). Nonlinear fourth order Taylor expansion of lattice Boltzmann schemes. *Asymptotic Analysis*, 127(4):297–337.

[Dubois et al., 2020a] Dubois, F., Graille, B., and Rao, S. R. (2020a). A notion of non-negativity preserving relaxation for a mono-dimensional three velocities scheme with relative velocity. *Journal of Computational Science*, 47:101181.

[Dubois and Lallemand, 2008] Dubois, F. and Lallemand, P. (2008). On lattice Boltzmann scheme, finite volumes and boundary conditions. *Progress in Computational Fluid Dynamics, an International Journal*, 8(1-4):11–24.

[Dubois and Lallemand, 2009] Dubois, F. and Lallemand, P. (2009). Towards higher order lattice Boltzmann schemes. *Journal of Statistical Mechanics: Theory and Experiment*, 2009(06):P06006.

[Dubois and Lallemand, 2011] Dubois, F. and Lallemand, P. (2011). Quartic parameters for acoustic applications of lattice Boltzmann scheme. *Computers & Mathematics with Applications*, 61(12):3404–3416.

[Dubois et al., 2015] Dubois, F., Lallemand, P., and Tekitek, M. M. (2015). Taylor expansion method for analyzing bounce-back boundary conditions for lattice Boltzmann method. *ESAIM: Proceedings and Surveys*, 52:25–46.

[Dubois et al., 2020b] Dubois, F., Lallemand, P., and Tekitek, M. M. (2020b). On anti bounce back boundary condition for lattice Boltzmann schemes. *Computers & Mathematics with Applications*, 79(3):555–575.

[Dummit and Foote, 2004] Dummit, D. S. and Foote, R. M. (2004). *Abstract Algebra*, volume 3. Wiley Hoboken.

[Dumont et al., 2013] Dumont, T., Duarte, M., Descombes, S., Dronne, M.-A., Massot, M., and Louvet, V. (2013). Simulation of human ischemic stroke in realistic 3D geometry. *Communications in Nonlinear Science and Numerical Simulation*, 18(6):1539–1557.

[Dupuis and Chopard, 2003] Dupuis, A. and Chopard, B. (2003). Theory and applications of an alternative lattice Boltzmann grid refinement algorithm. *Physical Review E*, 67(6):066707.

[Durran, 2013] Durran, D. R. (2013). *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*, volume 32. Springer Science & Business Media.

[Eitel-Amor et al., 2013] Eitel-Amor, G., Meinke, M., and Schröder, W. (2013). A lattice-Boltzmann method with hierarchically refined meshes. *Computers & Fluids*, 75:127–139.

[Engels et al., 2021] Engels, T., Schneider, K., Reiss, J., and Farge, M. (2021). A Wavelet-Adaptive Method for Multiscale Simulation of Turbulent Flows in Flying Insects . *Communications in Computational Physics*, 30(4):1118–1149.

[Eymard et al., 2000] Eymard, R., Gallouët, T., and Herbin, R. (2000). Finite volume methods. *Handbook of Numerical Analysis*, 7:713–1018.

[Fakhari et al., 2016] Fakhari, A., Geier, M., and Lee, T. (2016). A mass-conserving lattice Boltzmann method with dynamic grid refinement for immiscible two-phase flows. *Journal of Computational Physics*, 315:434–457.

[Fakhari and Lee, 2014] Fakhari, A. and Lee, T. (2014). Finite-difference lattice Boltzmann method with a block-structured adaptive-mesh-refinement technique. *Physical Review E*, 89(3):033310.

[Fakhari and Lee, 2015] Fakhari, A. and Lee, T. (2015). Numerics of the lattice Boltzmann method on nonuniform grids: standard LBM and finite-difference LBM. *Computers & Fluids*, 107:205–213.

[Farag et al., 2021] Farag, G., Zhao, S., Chiavassa, G., and Boivin, P. (2021). Consistency study of lattice-Boltzmann schemes macroscopic limit. *Physics of Fluids*, 33(3):037101.

[Feldhusen et al., 2016] Feldhusen, K., Deiterding, R., and Wagner, C. (2016). A dynamically adaptive lattice Boltzmann method for thermal convection problems. *International Journal of Applied Mathematics and Computer Science*, 26(4):735–747.

[Feng et al., 2020] Feng, Y., Guo, S., Jacob, J., and Sagaut, P. (2020). Grid refinement in the three-dimensional hybrid recursive regularized lattice Boltzmann method for compressible aerodynamics. *Physical Review E*, 101(6):063302.

[Février, 2014] Février, T. (2014). *Extension et analyse des schémas de Boltzmann sur réseau: les schémas à vitesse relative*. PhD thesis, Université Paris Sud-Paris XI.

[Filippova and Hänel, 1998] Filippova, O. and Hänel, D. (1998). Grid refinement for lattice-BGK models. *Journal of Computational Physics*, 147(1):219–228.

[Fliess and Mounier, 1998] Fliess, M. and Mounier, H. (1998). Controllability and observability of linear delay systems: an algebraic approach. *ESAIM: Control, Optimisation and Calculus of Variations*, 3:301–314.

[Forster, 2016] Forster, C. J. (2016). *Parallel wavelet-adaptive direct numerical simulation of multiphase flows with phase-change*. PhD thesis, Georgia Institute of Technology.

[Foti et al., 2020] Foti, D., Giorno, S., and Duraisamy, K. (2020). An adaptive mesh refinement approach based on optimal sparse sensing. *Theoretical and Computational Fluid Dynamics*, 34(4):457–482.

[Frisch et al., 1986] Frisch, U., Hasslacher, B., and Pomeau, Y. (1986). Lattice-Gas Automata for the Navier-Stokes Equation. *Physical Review Letters*, 56:1505–1508.

[Fučík and Straka, 2021] Fučík, R. and Straka, R. (2021). Equivalent finite difference and partial differential equations for the lattice Boltzmann method. *Computers & Mathematics with Applications*, 90:96–103.

[Gatignol, 1975] Gatignol, R. (1975). Theorie cinetique d'un gaz a repartition discrete de vitesses, in: Lecture Notes in Phys., Vol. 36, Springer-Verlag, 1975.

[Geier et al., 2006] Geier, M., Greiner, A., and Korvink, J. G. (2006). Cascaded digital lattice Boltzmann automata for high Reynolds number flow. *Physical Review E*, 73(6):066705.

[Gendre et al., 2017] Gendre, F., Ricot, D., Fritz, G., and Sagaut, P. (2017). Grid refinement for aeroacoustics in the lattice Boltzmann method: A directional splitting approach. *Physical Review E*, 96(2):023311.

[Ginzbourg and Adler, 1994] Ginzbourg, I. and Adler, P. (1994). Boundary flow condition analysis for the three-dimensional lattice Boltzmann model. *Journal de Physique II*, 4(2):191–214.

[Ginzburg, 2005] Ginzburg, I. (2005). Generic boundary conditions for lattice Boltzmann models and their application to advection and anisotropic dispersion equations. *Advances in Water Resources*, 28(11):1196–1216.

[Ginzburg, 2009] Ginzburg, I. (2009). *Une variation sur les propriétés magiques de modèles de Boltzmann pour l'écoulement microscopique et macroscopique*. Habilitation à diriger des recherches, Thèse d'Habilitation à diriger des recherches Spécialité Sciences pour l'ingénieur, Université Pierre et Marie Curie Paris.

[Ginzburg et al., 2023] Ginzburg, I., Silva, G., Marson, F., Chopard, B., and Latt, J. (2023). Unified directional parabolic-accurate lattice Boltzmann boundary schemes for grid-rotated narrow gaps and curved walls in creeping and inertial fluid flows. *Physical Review E*, 107(2):025303.

[Ginzburg et al., 2008a] Ginzburg, I., Verhaeghe, F., and d'Humières, D. (2008a). Study of simple hydrodynamic solutions with the two-relaxation-times lattice Boltzmann scheme. *Communications in Computational Physics*, 3(3):519–581.

[Ginzburg et al., 2008b] Ginzburg, I., Verhaeghe, F., and d'Humières, D. (2008b). Two-relaxation-time lattice Boltzmann scheme: About parametrization, velocity, pressure and mixed boundary conditions. *Communications in Computational Physics*, 3(2):427–478.

[Giusti and Williams, 1984] Giusti, E. and Williams, G. H. (1984). *Minimal surfaces and functions of bounded variation*, volume 80. Springer.

[Godlewski and Raviart, 1991] Godlewski, E. and Raviart, P.-A. (1991). *Hyperbolic systems of conservation laws*. Number 3-4. Ellipses.

[Godlewski and Raviart, 1996] Godlewski, E. and Raviart, P.-A. (1996). *Numerical approximation of hyperbolic systems of conservation laws*, volume 118. Springer.

[Godlewski and Raviart, 2013] Godlewski, E. and Raviart, P.-A. (2013). *Numerical approximation of hyperbolic systems of conservation laws*, volume 118. Springer Science & Business Media.

[Gottlieb et al., 2009] Gottlieb, S., Ketcheson, D. I., and Shu, C.-W. (2009). High order strong stability preserving time discretizations. *Journal of Scientific Computing*, 38(3):251–289.

[Graham, 1981] Graham, A. (1981). *Kronecker Products and Matrix Calculus: with Applications*. Ellis Horwood Limited.

[Graille, 2014] Graille, B. (2014). Approximation of mono-dimensional hyperbolic systems: A lattice Boltzmann scheme as a relaxation method. *Journal of Computational Physics*, 266:74–88.

[Graille et al., 2009] Graille, B., Magin, T. E., and Massot, M. (2009). Kinetic theory of plasmas: translational energy. *Mathematical Models and Methods in Applied Sciences*, 19(04):527–599.

[Guittet et al., 2015] Guittet, A., Theillard, M., and Gibou, F. (2015). A stable projection method for the incompressible Navier–Stokes equations on arbitrary geometries and adaptive Quad/Octrees. *Journal of Computational Physics*, 292:215–238.

[Guo et al., 2002] Guo, Z., Zheng, C., and Shi, B. (2002). An extrapolation method for boundary conditions in lattice Boltzmann method. *Physics of Fluids*, 14(6):2007–2010.

[Gustafsson et al., 1995] Gustafsson, B., Kreiss, H.-O., and Oliger, J. (1995). *Time dependent problems and difference methods*, volume 24. John Wiley & Sons.

[Gustafsson et al., 1972] Gustafsson, B., Kreiss, H.-O., and Sundström, A. (1972). Stability theory of difference approximations for mixed initial boundary value problems. II. *Mathematics of Computation*, 26(119):649–686.

[Hardy et al., 1973] Hardy, J., Pomeau, Y., and De Pazzis, O. (1973). Time evolution of a two-dimensional model system. I. Invariant states and time correlation functions. *Journal of Mathematical Physics*, 14(12):1746–1759.

[Harten, 1984] Harten, A. (1984). On a class of high resolution total-variation-stable finite-difference schemes. *SIAM Journal on Numerical Analysis*, 21(1):1–23.

[Harten, 1993] Harten, A. (1993). Discrete multi-resolution analysis and generalized wavelets. *Applied Numerical Mathematics*, 12(1-3):153–192.

[Harten, 1994] Harten, A. (1994). Adaptive multiresolution schemes for shock computations. *Journal of Computational Physics*, 115(2):319–338.

[Harten, 1995] Harten, A. (1995). Multiresolution algorithms for the numerical solution of hyperbolic conservation laws. *Communications on Pure and Applied Mathematics*, 48(12):1305–1342.

[Harten et al., 1987] Harten, A., Engquist, B., Osher, S., and Chakravarthy, S. R. (1987). Uniformly high order accurate essentially non-oscillatory schemes, III. In *Upwind and high-resolution schemes*, pages 218–290. Springer.

[Harten et al., 1976] Harten, A., Hyman, J. M., Lax, P. D., and Keyfitz, B. (1976). On finite-difference approximations and entropy conditions for shocks. *Communications on Pure and Applied Mathematics*, 29(3):297–322.

[Harten and Lax, 1981] Harten, A. and Lax, P. D. (1981). A random choice finite difference scheme for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 18(2):289–315.

[He et al., 1998] He, X., Shan, X., and Doolen, G. D. (1998). Discrete Boltzmann equation model for nonideal gases. *Physical Review E*, 57(1):R13.

[Hendricks et al., 2008] Hendricks, E., Jannerup, O., and Sørensen, P. H. (2008). *Linear systems control: deterministic and stochastic methods*. Springer.

[Hénon, 1987] Hénon, M. (1987). Viscosity of a lattice gas. *Lattice Gas Methods for Partial Differential Equations*, pages 179–207.

[Higuera et al., 1989] Higuera, F., Succi, S., and Benzi, R. (1989). Lattice gas dynamics with enhanced collisions. *Europhysics Letters*, 9(4):345.

[Higuera and Jiménez, 1989] Higuera, F. J. and Jiménez, J. (1989). Boltzmann approach to lattice gas simulations. *Europhysics Letters*, 9(7):663.

[Hopf, 1950] Hopf, E. (1950). The partial differential equation $u_t + uu_x = \mu u_{xx}$. *Communications on Pure and Applied Mathematics*, 3(3):201–230.

[Horn and Johnson, 2012] Horn, R. A. and Johnson, C. R. (2012). *Matrix analysis*. Cambridge University Press.

[Horstmann, 2018] Horstmann, J. (2018). *Hybrid numerical method based on the lattice Boltzmann approach with application to non-uniform grids*. PhD thesis, Université de Lyon.

[Hou, 1998] Hou, S.-H. (1998). Classroom Note: A Simple Proof of the Leverrier–Faddeev Characteristic Polynomial Algorithm. *SIAM Review*, 40(3):706–709.

[Hovhannisyan and Müller, 2010] Hovhannisyan, N. and Müller, S. (2010). On the stability of fully adaptive multiscale schemes for conservation laws using approximate flux and source reconstruction strategies. *IMA Journal of Numerical Analysis*, 30(4):1256–1295.

[Huang et al., 2015a] Huang, H., Sukop, M., and Lu, X. (2015a). *Multiphase lattice Boltzmann methods: Theory and application*. John Wiley & Sons.

[Huang et al., 2015b] Huang, J., Wu, H., and Yong, W.-A. (2015b). On initial conditions for the lattice Boltzmann method. *Communications in Computational Physics*, 18(2):450–468.

[Huang, 1987] Huang, K. (1987). *Statistical Mechanics*. John Wiley & Sons, 2 edition.

[Hundsdorfer et al., 2009] Hundsdorfer, W., Mozartova, A., and Spijker, M. (2009). Stepsize conditions for boundedness in numerical initial value problems. *SIAM Journal on Numerical Analysis*, 47(5):3797–3819.

[Hundsdorfer and Ruuth, 2006] Hundsdorfer, W. and Ruuth, S. (2006). On monotonicity and boundedness properties of linear multistep methods. *Mathematics of Computation*, 75(254):655–672.

[Hundsdorfer et al., 2003] Hundsdorfer, W., Ruuth, S. J., and Spiteri, R. J. (2003). Monotonicity-preserving linear multistep methods. *SIAM Journal on Numerical Analysis*, 41(2):605–623.

[Jaming and Malinnikova, 2016] Jaming, P. and Malinnikova, E. (2016). An uncertainty principle and sampling inequalities in Besov spaces. *Journal of Fourier Analysis and Applications*, 22:768–786.

[Jin and Xin, 1995] Jin, S. and Xin, Z. (1995). The relaxation schemes for systems of conservation laws in arbitrary space dimensions. *Communications on Pure and Applied mathematics*, 48(3):235–276.

[Johnson, 2002] Johnson, W. P. (2002). The curious history of Faà di Bruno's formula. *The American Mathematical Monthly*, 109(3):217–234.

[Junk et al., 2005] Junk, M., Klar, A., and Luo, L.-S. (2005). Asymptotic analysis of the lattice Boltzmann equation. *Journal of Computational Physics*, 210(2):676–704.

[Junk and Rheinlander, 2008] Junk, M. and Rheinlander, M. (2008). Regular and multiscale expansions of a lattice Boltzmann method. *Progress in Computational Fluid Dynamics, an International Journal*, 8(1-4):25–37.

[Junk and Yang, 2009] Junk, M. and Yang, Z. (2009). Convergence of lattice Boltzmann methods for Navier–Stokes flows in periodic and bounded domains. *Numerische Mathematik*, 112(1):65–87.

[Junk and Yang, 2015] Junk, M. and Yang, Z. (2015). L^2 convergence of the lattice Boltzmann method for one dimensional convection-diffusion-reaction equations. *Communications in Computational Physics*, 17(5):1225–1245.

[Junk and Yong, 2003] Junk, M. and Yong, W.-A. (2003). Rigorous Navier–Stokes limit of the lattice Boltzmann equation. *Asymptotic Analysis*, 35(2):165–185.

[Junk and Yong, 2009] Junk, M. and Yong, W.-A. (2009). Weighted L^2-Stability of the Lattice Boltzmann Method. *SIAM Journal on Numerical Analysis*, 47(3):1651–1665.

[Jury, 1964] Jury, E. I. (1964). *Theory and Application of the z-Transform Method*. Krieger Publishing Co.

[Kandhai et al., 2000] Kandhai, D., Soll, W., Chen, S., Hoekstra, A., and Sloot, P. (2000). Finite-difference lattice-BGK methods on nested grids. *Computer Physics Communications*, 129(1-3):100–109.

[Kassel, 1995] Kassel, C. (1995). *Quantum Groups*. Graduate Texts in Mathematics. Springer-Verlag New York, 1 edition.

[Kelly et al., 1994] Kelly, S. E., Kon, M. A., and Raphael, L. A. (1994). Pointwise convergence of wavelet expansions. *Bulletin of The American Mathematical Society*, 30(1):87–94.

[Kiris et al., 2018] Kiris, C. C., Stich, D., Housman, J. A., Kocheemoolayil, J. G., Barad, M. F., and Cadieux, F. (2018). Application of Lattice Boltzmann and Navier-Stokes methods to NASA's wall mounted hump. In *2018 Fluid Dynamics Conference*, page 3855.

[Kraaijevanger, 1994] Kraaijevanger, J. F. B. M. (1994). Two counterexamples related to the Kreiss matrix theorem. *BIT Numerical Mathematics*, 34:113–119.

[Krüger et al., 2017] Krüger, T., Kusumaatmaja, H., Kuzmin, A., Shardt, O., Silva, G., and Viggen, E. M. (2017). The lattice Boltzmann method. *Springer International Publishing*, 10(978-3):4–15.

[Kuzmin et al., 2011] Kuzmin, A., Ginzburg, I., and Mohamad, A. (2011). The role of the kinetic parameter in the stability of two-relaxation-time advection–diffusion lattice Boltzmann schemes. *Computers & Mathematics with Applications*, 61(12):3417–3442.

[Kuznik et al., 2013] Kuznik, F., Luo, L.-S., and Krafczyk, M. (2013). Mesoscopic Methods in Engineering and Science. *Computers & Mathematics with Applications*, 65(6):813–814.

[Ladd, 1994] Ladd, A. J. (1994). Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. *Journal of Fluid Mechanics*, 271:285–309.

[Lagrava, 2012] Lagrava, S. D. W. (2012). *Revisiting grid refinement algorithms for the lattice Boltzmann method*. PhD thesis, University of Geneva.

[Lallemand and Luo, 2000] Lallemand, P. and Luo, L.-S. (2000). Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability. *Physical Review E*, 61(6):6546.

[Lamby et al., 2005] Lamby, P., Müller, S., and Stiriba, Y. (2005). Solution of shallow water equations using fully adaptive multiscale schemes. *International Journal for Numerical Methods in Fluids*, 49(4):417–437.

[Landajuela, 2011] Landajuela, M. (2011). Burgers equation. *BCAM Internship report: Basque Center for Applied Mathematics*.

[Lang, 2002] Lang, S. (2002). *Algebra*. Graduate Texts in Mathematics. Springer-Verlag New York, 3 edition.

[Lax and Liu, 1998] Lax, P. D. and Liu, X.-D. (1998). Solution of two-dimensional Riemann problems of gas dynamics by positive schemes. *SIAM Journal on Scientific Computing*, 19(2):319–340.

[Lax and Richtmyer, 1956] Lax, P. D. and Richtmyer, R. D. (1956). Survey of the stability of linear finite difference equations. *Communications on Pure and Applied Mathematics*, 9(2):267–293.

[Lecointre, 2022] Lecointre, L. (2022). *Hydrogen flame acceleration in non-uniform mixtures*. PhD thesis, Université Paris-Saclay.

[Lemarié-Rieusset, 1996] Lemarié-Rieusset, P. G. (1996). Some remarks on orthogonal and bi-orthogonal wavelets. *Computation and Applied Mathematics*, 15:125–138.

[LeVeque, 2002] LeVeque, R. J. (2002). *Finite volume methods for hyperbolic problems*, volume 31. Cambridge University Press.

[LeVeque and Trefethen, 1984] LeVeque, R. J. and Trefethen, L. N. (1984). On the resolvent condition in the Kreiss matrix theorem. *BIT Numerical Mathematics*, 24(4):584–591.

[Lin and Lai, 2000] Lin, C.-L. and Lai, Y. G. (2000). Lattice Boltzmann method on composite grids. *Physical Review E*, 62(2):2219.

[Lin et al., 2021] Lin, Y., Hong, N., Shi, B., and Chai, Z. (2021). Multiple-relaxation-time lattice Boltzmann model-based four-level finite-difference scheme for one-dimensional diffusion equations. *Physical Review E*, 104(1):015312.

[Liska and Wendroff, 2003] Liska, R. and Wendroff, B. (2003). Comparison of several difference schemes on 1D and 2D test problems for the Euler equations. *SIAM Journal on Scientific Computing*, 25(3):995–1017.

[Mallat, 1989] Mallat, S. G. (1989). Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. *Transactions of the American Mathematical Society*, 315(1):69–87.

[Marié et al., 2009] Marié, S., Ricot, D., and Sagaut, P. (2009). Comparison between lattice Boltzmann method and Navier–Stokes high order schemes for computational aeroacoustics. *Journal of Computational Physics*, 228(4):1056–1070.

[Marson et al., 2021] Marson, F., Thorimbert, Y., Chopard, B., Ginzburg, I., and Latt, J. (2021). Enhanced single-node lattice Boltzmann boundary condition for fluid flows. *Physical Review E*, 103(5):053308.

[Martínez et al., 1994] Martínez, D. O., Chen, S., and Matthaeus, W. H. (1994). Lattice Boltzmann magnetohydrodynamics. *Physics of Plasmas*, 1(6):1850–1867.

[Mazzeo and Coveney, 2008] Mazzeo, M. D. and Coveney, P. V. (2008). HemeLB: A high performance parallel lattice-Boltzmann code for large scale fluid flow in complex geometries. *Computer Physics Communications*, 178(12):894–914.

[McNamara and Zanetti, 1988] McNamara, G. R. and Zanetti, G. (1988). Use of the Boltzmann equation to simulate lattice-gas automata. *Physical Review Letters*, 61(20):2332.

[Milies et al., 2002] Milies, C. P., Sehgal, S. K., and Sehgal, S. (2002). *An introduction to group rings*, volume 1. Springer Science & Business Media.

[Miller, 1971] Miller, J. J. (1971). On the location of zeros of certain classes of polynomials with applications to numerical analysis. *IMA Journal of Applied Mathematics*, 8(3):397–406.

[Miller, 1960] Miller, K. S. (1960). *An Introduction to the Calculus of Finite Differences and Difference Equations*. Dover Publications.

[Milne-Thomson, 1933] Milne-Thomson, L. M. (1933). *The calculus of finite differences*. MacMillan and Co.

[Mohamad and Kuzmin, 2012] Mohamad, A. and Kuzmin, A. (2012). The Soret effect with the D1Q2 and D2Q4 lattice Boltzmann model. *International Journal of Nonlinear Sciences and Numerical Simulation*, 13(3-4):289–293.

[Mohamad and Succi, 2009] Mohamad, A. and Succi, S. (2009). A note on equilibrium boundary conditions in lattice Boltzmann fluid dynamic simulations. *The European Physical Journal Special Topics*, 171(1):213–221.

[Monforte and Kauers, 2013] Monforte, A. A. and Kauers, M. (2013). Formal Laurent series in several variables. *Expositiones Mathematicae*, 31(4):350–367.

[Müller, 2002] Müller, S. (2002). *Adaptive multiscale schemes for conservation laws*, volume 27. Springer Science & Business Media.

[Naddei et al., 2019] Naddei, F., de la Llave Plata, M., Couaillier, V., and Coquel, F. (2019). A comparison of refinement indicators for p-adaptive simulations of steady and unsteady flows using discontinuous Galerkin methods. *Journal of Computational Physics*, 376:508–533.

[Najafi-Yazdi and Mongeau, 2012] Najafi-Yazdi, A. and Mongeau, L. (2012). An absorbing boundary condition for the lattice Boltzmann method based on the perfectly matched layer. *Computers & Fluids*, 68:203–218.

[Narechania et al., 2017] Narechania, N. M., Freret, L. V., and Groth, C. P. (2017). Block-based anisotropic AMR with A Posteriori adjoint-based error estimation for three-dimensional inviscid and viscous flows. In *23rd AIAA Computational Fluid Dynamics Conference*, page 4113.

[N'guessan, 2020] N'guessan, M.-A. (2020). *Space adaptive methods with error control based on adaptive multiresolution for the simulation of low-Mach reactive flows*. PhD thesis, Université Paris-Saclay.

[N'Guessan et al., 2021] N'Guessan, M.-A., Massot, M., Series, L., and Tenaud, C. (2021). High order time integration and mesh adaptation with error control for incompressible Navier-Stokes and scalar transport resolution on dual grids. *Journal of Computational and Applied Mathematics*, 387:112542.

[Niven, 1969] Niven, I. (1969). Formal power series. *The American Mathematical Monthly*, 76(8):871–889.

[O'Malley, 1991] O'Malley, R. E. (1991). *Singular perturbation methods for ordinary differential equations*, volume 89. Springer.

[Pan et al., 2006] Pan, C., Luo, L.-S., and Miller, C. T. (2006). An evaluation of lattice Boltzmann schemes for porous medium flow simulation. *Computers & Fluids*, 35(8-9):898–909.

[Pan et al., 2018] Pan, S., Hu, X., and Adams, N. (2018). Positivity-preserving method for multi-resolution simulations of compressible flows. *arXiv preprint arXiv:1807.07053*.

[Platkowski and Illner, 1988] Platkowski, T. and Illner, R. (1988). Discrete velocity models of the Boltzmann equation: a survey on the mathematical aspects of the theory. *SIAM Review*, 30(2):213–255.

[Qian et al., 1992] Qian, Y.-H., d'Humières, D., and Lallemand, P. (1992). Lattice BGK models for Navier-Stokes equation. *Europhysics Letters*, 17(6):479.

[Qian and Zhou, 2000] Qian, Y.-H. and Zhou, Y. (2000). Higher-order dynamics in lattice-based models using the Chapman-Enskog method. *Physical Review E*, 61(2):2103.

[Ray et al., 2007] Ray, J., Kennedy, C. A., Lefantzi, S., and Najm, H. N. (2007). Using high-order methods on adaptively refined block-structured meshes: derivatives, interpolations, and filters. *SIAM Journal on Scientific Computing*, 29(1):139–181.

[Rheinländer, 2010] Rheinländer, M. (2010). On the stability structure for lattice Boltzmann schemes. *Computers & Mathematics with Applications*, 59(7):2150–2167.

[Rohde et al., 2006] Rohde, M., Kandhai, D., Derksen, J. J., and Van den Akker, H. E. A. (2006). A generic, mass conservative local grid refinement technique for lattice-Boltzmann schemes. *International Journal for Numerical Methods in Fluids*, 51(4):439–468.

[Roman, 2005] Roman, S. (2005). *The umbral calculus*. Springer.

[Rossi et al., 2005] Rossi, N., Ubertini, S., Bella, G., and Succi, S. (2005). Unstructured lattice Boltzmann method in three dimensions. *International Journal for Numerical Methods in Fluids*, 49(6):619–633.

[Rota et al., 1973] Rota, G.-C., Kahaner, D., and Odlyzko, A. (1973). On the foundations of combinatorial theory. VIII. Finite operator calculus. *Journal of Mathematical Analysis and Applications*, 42(3):684–760.

[Roussel and Schneider, 2005] Roussel, O. and Schneider, K. (2005). An adaptive multiresolution method for combustion problems: application to flame ball–vortex interaction. *Computers & Fluids*, 34(7):817–831.

[Roussel et al., 2003] Roussel, O., Schneider, K., Tsigulin, A., and Bockhorn, H. (2003). A conservative fully adaptive multiresolution algorithm for parabolic PDEs. *Journal of Computational Physics*, 188(2):493–523.

[Saad, 1989] Saad, Y. (1989). Overview of Krylov subspace methods with applications to control problems. Technical report.

[Samarskii, 2001] Samarskii, A. A. (2001). *The theory of difference schemes*. CRC Press.

[Sanders, 1983] Sanders, R. (1983). On convergence of monotone finite difference schemes with variable spatial differencing. *Mathematics of Computation*, 40(161):91–106.

[Serre, 1999] Serre, D. (1999). *Systems of Conservation Laws 1: Hyperbolicity, entropies, shock waves*. Cambridge University Press.

[Simonis et al., 2020] Simonis, S., Frank, M., and Krause, M. J. (2020). On relaxation systems and their relation to discrete velocity Boltzmann models for scalar advection–diffusion equations. *Philosophical Transactions of the Royal Society A*, 378(2175):20190400.

[Sod, 1978] Sod, G. A. (1978). A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1–31.

[Soni et al., 2017] Soni, V., Roussel, O., and Hadjadj, A. (2017). On the accuracy and efficiency of point-value multiresolution algorithms for solving scalar wave and Euler equations. *Journal of Computational and Applied Mathematics*, 323:159–175.

[Spijker, 2007] Spijker, M. (2007). Stepsize conditions for general monotonicity in numerical initial value problems. *SIAM Journal on Numerical Analysis*, 45(3):1226–1245.

[Stein and Shakarchi, 2011] Stein, E. M. and Shakarchi, R. (2011). *Fourier Analysis: An Introduction*, volume 1. Princeton University Press.

[Sterling and Chen, 1996] Sterling, J. D. and Chen, S. (1996). Stability analysis of lattice Boltzmann methods. *Journal of Computational Physics*, 123(1):196–206.

[Stewart, 1998] Stewart, G. (1998). On the adjugate matrix. *Linear Algebra and its Applications*, 283(1-3):151–164.

[Strikwerda, 2004] Strikwerda, J. C. (2004). *Finite difference schemes and partial differential equations*. SIAM.

[Strikwerda and Wade, 1994] Strikwerda, J. C. and Wade, B. A. (1994). A survey of the Kreiss matrix theorem for power bounded families of matrices and its extensions. Technical report, University of Wisconsin-Madison Department of Computer Sciences.

[Suga, 2010] Suga, S. (2010). An accurate multi-level finite difference scheme for 1D diffusion equations derived from the lattice Boltzmann method. *Journal of Statistical Physics*, 140(3):494–503.

[Sword, 1956] Sword, C. (1956). Numerical dispersion and attenuation in the M3 modeling system. *Standford Exploration Project*, Technical Report SEP-561.

[Tadmor, 1984] Tadmor, E. (1984). Numerical viscosity and the entropy condition for conservative difference schemes. *Mathematics of Computation*, 43(168):369–381.

[Tinney and Walker, 1967] Tinney, W. F. and Walker, J. W. (1967). Direct solutions of sparse network equations by optimally ordered triangular factorization. *Proceedings of the IEEE*, 55(11):1801–1809.

[Tomczak and Szafran, 2019] Tomczak, T. and Szafran, R. G. (2019). A new GPU implementation for lattice-Boltzmann simulations on sparse geometries. *Computer Physics Communications*, 235:258–278.

[Toro, 2009] Toro, E. F. (2009). *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. Springer Science & Business Media, third edition.

[Trefethen, 1982] Trefethen, L. N. (1982). *Wave propagation and stability for finite difference schemes*. Stanford University.

[Trefethen, 1984] Trefethen, L. N. (1984). Instability of difference models for hyperbolic initial boundary value problems. *Communications on Pure and Applied Mathematics*, 37(3):329–367.

[Trefethen, 1996] Trefethen, L. N. (1996). *Finite Difference and Spectral Methods for Ordinary and Partial Differential Equations*. unpublished text.

[Ubertini and Succi, 2005] Ubertini, S. and Succi, S. (2005). Recent advances of lattice Boltzmann techniques on unstructured grids. *Progress in Computational Fluid Dynamics, an International Journal*, 5(1-2):85–96.

[Van Leemput et al., 2009] Van Leemput, P., Rheinländer, M., and Junk, M. (2009). Smooth initialization of lattice Boltzmann schemes. *Computers & Mathematics with Applications*, 58(5):867–882.

[Vidal et al., 2010] Vidal, D., Roy, R., and Bertrand, F. (2010). On improving the performance of large parallel lattice Boltzmann flow simulations in heterogeneous porous media. *Computers & Fluids*, 39(2):324–337.

[Wargnier, 2019] Wargnier, Q. (2019). *Mathematical modeling and simulation of non-equilibrium plasmas: application to magnetic reconnection in the Sun atmosphere*. PhD thesis, Université Paris Saclay.

[Warming and Hyett, 1974] Warming, R. F. and Hyett, B. (1974). The modified equation approach to the stability and accuracy analysis of finite-difference methods. *Journal of Computational Physics*, 14(2):159–179.

[Wissink, 2001] Wissink, A. (2001). Large scale structured AMR calculations using the SAMRAI framework. *SC01 Proceedings, 2001*.

[Wu et al., 1990] Wu, J., Zhu, J., Szmelter, J., and Zienkiewicz, O. (1990). Error estimation and adaptivity in Navier-Stokes incompressible flows. *Computational Mechanics*, 6(4):259–270.

[Yong et al., 2016] Yong, W.-A., Zhao, W., Luo, L.-S., et al. (2016). Theory of the lattice Boltzmann method: Derivation of macroscopic equations via the Maxwell iteration. *Physical Review E*, 93(3):033310.

[Yu et al., 2002] Yu, D., Mei, R., and Shyy, W. (2002). A multi-block lattice Boltzmann method for viscous fluid flows. *International Journal for Numerical Methods in Fluids*, 39(2):99–120.

[Zhang et al., 2019] Zhang, M., Zhao, W., and Lin, P. (2019). Lattice Boltzmann method for general convection-diffusion equations: MRT model and boundary schemes. *Journal of Computational Physics*, 389:147–163.

[Zhang et al., 2021] Zhang, W., Myers, A., Gott, K., Almgren, A., and Bell, J. (2021). AMReX: Block-structured adaptive mesh refinement for multiphysics applications. *The International Journal of High Performance Computing Applications*, 35(6):508–526.

[Zhao and Yong, 2017] Zhao, W. and Yong, W.-A. (2017). Maxwell iteration for the lattice Boltzmann method with diffusive scaling. *Physical Review E*, 95(3):033311.

[Zhong et al., 2006] Zhong, L., Feng, S., Dong, P., and Gao, S. (2006). Lattice Boltzmann schemes for the nonlinear Schrödinger equation. *Physical Review E*, 74(3):036704.

[Zwillinger, 2018] Zwillinger, D. (2018). *CRC standard mathematical tables and formulas*. Chapman and Hall - CRC.

Appendix A

Various computations

A.1 Stability of the D_1Q_2 scheme of Section 10.3.1

There are several ways of checking the roots of amplification polynomial of the corresponding bulk scheme for Section 10.3.1. In this case, we can proceed directly by solving the characteristic equation or by using the procedure by [Graille, 2014]. Thanks to its generality, we here present the computations using the technique [Miller, 1971] given in Theorem 7.7.2. The amplification polynomial reads $\hat{\Phi}(\xi\Delta x, z) = z^2 + ((s_2-2)\cos(\xi\Delta x) + i s_2\epsilon_2 \sin(\xi\Delta x))z + (1-s_2)$, where $\xi \in [-\pi/\Delta x, \pi/\Delta x]$. We set $\hat{\Phi}_2 \equiv \hat{\Phi}$. We have that

$$\hat{\Phi}_2^\star(\xi\Delta x, z) := z^2\hat{\Phi}_2(z^{-1}, -\xi\Delta x) = (1-s_2)z^2 + ((s_2-2)\cos(\xi\Delta x) - i s_2\epsilon_2 \sin(\xi\Delta x))z + 1.$$

- Let $s_2 \in]0,2[$. A first condition to bound the roots of $\hat{\Phi}_2(\xi\Delta x, z)$ according to Theorem 7.7.3 regardless of the frequency is that $|\hat{\Phi}_2(\xi\Delta x, 0)| < |\hat{\Phi}_2^\star(\xi\Delta x, 0)|$, which yields the condition $|1-s_2| < 1$, hence $0 < s_2 < 2$. Then, we compute

$$\begin{aligned}\hat{\Phi}_1(\xi\Delta x, z) := z^{-1}(\hat{\Phi}_2^\star(\xi\Delta x, 0)\hat{\Phi}_2(\xi\Delta x, z) - \hat{\Phi}_2(\xi\Delta x, 0)\hat{\Phi}_2^\star(\xi\Delta x, z)) \\ = s_2(2-s_2)(z - \cos(\xi\Delta x) + i\epsilon_2 \sin(\xi\Delta x)).\end{aligned}$$

  The final condition to check is that the root of $\hat{\Phi}_1(\xi\Delta x, z)$ is bounded by one in modulus for any frequency. This is $\cos^2(\xi\Delta x) + \epsilon_2^2 \sin^2(\xi\Delta x) = 1 + (\epsilon_2^2 - 1)\sin^2(\xi\Delta x) \leq 1$ taking place for any $\xi \in [-\pi/\Delta x, \pi/\Delta x]$ if and only if $\epsilon_2^2 \leq 1$.
- Let $s_2 = 2$. In this case $\hat{\Phi}_1(\xi\Delta x, z) \equiv 0$. We then have to use the second condition from Theorem 7.7.2, hence we check
$$\frac{d\hat{\Phi}_2(\xi\Delta x, z)}{dz} = 2z + 2i\epsilon_2 \sin(\xi\Delta x)),$$
  which unique root should be strictly in the unit circle for any frequency $\xi \in [-\pi/\Delta x, \pi/\Delta x]$. This is achieved by $|\epsilon_2| < 1$.

A.2 Derivation of the forward centered initialisation schemes for the D_1Q_2 of Section 10.3.1

We can first unsuccessfully attempt to obtain a forward centered scheme as initialisation scheme, using a local initialisation of the conserved moment, that is $w_1 = 1$ and prepared initialisation of the non-conserved one, thus $w_2 \in D$. Using the notation (10.21), this corresponds to find a compactly supported solution of the following infinite system

$$\begin{aligned}\ldots, \quad w_{2,1} - w_{2,3} = 0, \quad w_{2,0} - w_{2,2} = -\frac{1-\epsilon_2+s_2\epsilon_2}{1-s_2}, \quad w_{2,-1} - w_{2,1} = \frac{2}{1-s_2}, \\ w_{2,-2} - w_{2,0} = -\frac{1+\epsilon_2-s_2\epsilon_2}{1-s_2}, \quad w_{2,-3} - w_{2,-1} = 0, \quad \ldots\end{aligned}$$

This problem cannot be solved by a compactly supported sequence, in particular, because of the median term. This would go back to perform a deconvolution in the ring of Finite Difference operators, which is not solvable because the operator $\mathsf{A}(\mathsf{x}_1)$ is not invertible in such ring. If we consider to work on a bounded domain with $N_x \in \mathbb{N}^*$ points and endow the shift operators with periodic boundary conditions [Van Leemput et al., 2009], some of these deconvolution problems become solvable at the price of dealing with non-compactly supported solutions, *i.e.* stemming from a full inverse of a sparse matrix. The previous problem can be seen as the one of inverting a circulant matrix, which eigenvalues are $\sigma_r = \exp(2\pi i(N_x-1)r/N_x) - \exp(2\pi i r/N_x)$ for $r \in [\![0, N_x[\![$. Since $\sigma_0 = 0$, the circulant matrix is not invertible. Therefore, even in the periodic setting, this procedure does not work. This can be interpreted—if we see the equilibria as a control on the system—as due to the lack of "reachability" of the system at hand, *cf.* [Brewer et al., 1986, Chapter 2]. Since the term $\mathsf{A}(\mathsf{x}_1)$ is not a unit, which causes the lack of reachability, it cannot be compensated by its inverse contained in the equilibrium to generate the desired initialisation scheme. This is why we are compelled to consider $\mathsf{w}_1 \in \mathsf{D}$ to obtain the requested forward centered scheme.

Considering a prepared initialisation for both moments, thus $\mathsf{w}_1, \mathsf{w}_2 \in \mathsf{D}$, several choices are possible to recover this scheme. The infinite system to solve reads

$$\begin{aligned}
&\dots\\
\frac{1+s_2\epsilon_2}{2}w_{1,1} + \frac{1-s_2}{2}w_{2,1} + \frac{1-s_2\epsilon_2}{2}w_{1,3} - \frac{1-s_2}{2}w_{2,3} &= 0,\\
\frac{1+s_2\epsilon_2}{2}w_{1,0} + \frac{1-s_2}{2}w_{2,0} + \frac{1-s_2\epsilon_2}{2}w_{1,2} - \frac{1-s_2}{2}w_{2,2} &= \frac{\epsilon_2}{2},\\
\frac{1+s_2\epsilon_2}{2}w_{1,-1} + \frac{1-s_2}{2}w_{2,-1} + \frac{1-s_2\epsilon_2}{2}w_{1,1} - \frac{1-s_2}{2}w_{2,1} &= 1,\\
\frac{1+s_2\epsilon_2}{2}w_{1,-2} + \frac{1-s_2}{2}w_{2,-2} + \frac{1-s_2\epsilon_2}{2}w_{1,0} - \frac{1-s_2}{2}w_{2,0} &= -\frac{\epsilon_2}{2},\\
\frac{1+s_2\epsilon_2}{2}w_{1,-3} + \frac{1-s_2}{2}w_{2,-3} + \frac{1-s_2\epsilon_2}{2}w_{1,-1} - \frac{1-s_2}{2}w_{2,-1} &= 0,\\
&\dots
\end{aligned}$$

In order to construct a (non-unique) solution, we first enforce the compactness: $w_{1,r} = w_{2,r} = 0$ for $|r| \geq 2$. From this, we obtain the finite system

$$\begin{aligned}
(1+s_2\epsilon_2)w_{1,1} + (1-s_2)w_{2,1} &= 0,\\
(1+s_2\epsilon_2)w_{1,0} + (1-s_2)w_{2,0} &= \epsilon_2,\\
(1+s_2\epsilon_2)w_{1,-1} + (1-s_2)w_{2,-1} + (1-s_2\epsilon_2)w_{1,1} - (1-s_2)w_{2,1} &= 2,\\
(1-s_2\epsilon_2)w_{1,0} - (1-s_2)w_{2,0} &= -\epsilon_2,\\
(1-s_2\epsilon_2)w_{1,-1} - (1-s_2)w_{2,-1} &= 0.
\end{aligned}$$

We then split the central equation using a parameter $\theta \in \mathbb{R}$, having $(1+s_2\epsilon_2)w_{1,-1} + (1-s_2)w_{2,-1} = \theta$ and $(1-s_2\epsilon_2)w_{1,1} - (1-s_2)w_{2,1} = 2-\theta$. Introducing the matrix

$$\boldsymbol{A} = \begin{bmatrix} 1+s_2\epsilon_2 & 1-s_2 \\ 1-s_2\epsilon_2 & s_2-1 \end{bmatrix},$$

we solve the systems $\boldsymbol{A}(w_{1,1}, w_{2,1})^{\mathsf{t}} = (0, 2-\theta)^{\mathsf{t}}$, $\boldsymbol{A}(w_{1,0}, w_{2,0})^{\mathsf{t}} = (\epsilon_2, -\epsilon_2)^{\mathsf{t}}$ and $\boldsymbol{A}(w_{1,-1}, w_{2,-1})^{\mathsf{t}} = (\theta, 0)^{\mathsf{t}}$, yielding

$$\begin{aligned}
w_{1,1} &= \frac{2-\theta}{2}, \qquad & w_{2,1} &= -\frac{(1+s_2\epsilon_2)(2-\theta)}{2(1-s_2)}, \qquad & w_{1,0} &= 0, \qquad & w_{2,0} &= \frac{\epsilon_2}{1-s_2},\\
w_{1,-1} &= \frac{\theta}{2}, \qquad & w_{2,-1} &= \frac{(1-s_2\epsilon_2)\theta}{2(1-s_2)}.
\end{aligned}$$

Unsurprisingly, these coefficients are defined for $s_2 \neq 1$, since otherwise there is no initialisation scheme to devise.

The only way to fulfill (10.22), (10.23) and (10.24) is to take $\theta = 1$, giving

$$w_{1,\pm1} = \frac{1}{2}, \qquad w_{2,\pm1} = \mp\frac{1 \pm s_2\epsilon_2}{2(1-s_2)}, \qquad w_{2,0} = \frac{\epsilon_2}{1-s_2}.$$

Allowing more non-vanishing coefficients and through a similar procedure, another possible choice to obtain the desired scheme would be

$$w_{1,\pm2} = \pm\frac{\epsilon_2}{2}, \qquad w_{1,\pm1} = \frac{1}{2}, \qquad w_{2,\pm2} = -\frac{\epsilon_2(1 \pm s_2\epsilon_2)}{2(1-s_2)}, \qquad w_{2,\pm1} = \mp\frac{1 \pm s_2\epsilon_2}{2(1-s_2)}.$$

A.3 Stability of the D_1Q_3 scheme of Section 10.3.2 when $s_2 + s_3 = 2$

Once more, we apply the technique by Theorem 7.7.2. The amplification polynomial reads $\hat{\Phi}(\xi\Delta x, z) = (z + (1 - s_2))(z^2 + ((s_2-2)(2\cos(\xi\Delta x)+1)/3 + (s_2-2)\epsilon_3(\cos(\xi\Delta x)-1)/3 + i s_2\epsilon_2 \sin(\xi\Delta x))z + (1-s_2))$ with $\xi \in [-\pi/\Delta x, \pi/\Delta x]$. The polynomial which roots need to be controlled is $\hat{\Psi}_2(\xi\Delta x, z) = z^2 + ((s_2-2)(2\cos(\xi\Delta x)+1)/3 + (s_2-2)\epsilon_3(\cos(\xi\Delta x) - 1)/3 + i s_2\epsilon_2 \sin(\xi\Delta x))z + (1-s_2)$. We have

$$\hat{\Psi}_2^\star(\xi\Delta x, z) = (1-s_2)z^2 + ((s_2-2)(2\cos(\xi\Delta x)+1)/3 + (s_2-2)\epsilon_3(\cos(\xi\Delta x)-1)/3 - i s_2\epsilon_2\sin(\xi\Delta x))z + 1.$$

- Let $s_2 \in]0,2[$. Checking the first condition $|\hat{\Psi}_2(\xi\Delta x, 0)| < |\hat{\Psi}_2^\star(\xi\Delta x, 0)|$ trivially gives $0 < s_2 < 2$, which is already fulfilled. Then we have

  $$\hat{\Psi}_1(\xi\Delta x, z) := s_2(2-s_2)(z - (2\cos(\xi\Delta x)+1)/3 - \epsilon_3(\cos(\xi\Delta x)-1)/3 + i\epsilon_2\sin(\xi\Delta x)).$$

  Checking that is unique root lies inside or on the unit circle boils down to check $((2\cos(\xi\Delta x)+1) + \epsilon_3(\cos(\xi\Delta x) - 1))^2/9 + \epsilon_2^2\sin^2(\xi\Delta x) \le 1$. Using the trigonometric identities $\cos(\xi\Delta x) = 1 - 2\sin^2(\xi\Delta x/2)$ and $\sin^2(\xi\Delta x) = 4\sin^2(\xi\Delta x/2)(1 - \sin^2(\xi\Delta x/2))$. Remark that $\sin^2(\xi\Delta x/2) \in [0,1]$, hence we obtain the condition

  $$\sin^4(\xi\Delta x/2)((\epsilon_3+2)^2/9 - \epsilon_2^2) + \sin^2(\xi\Delta x/2)(-(\epsilon_3+2)/3 + \epsilon_2^2) \le 0, \qquad \forall \sin^2(\xi\Delta x/2) \in [0,1].$$

  It is fulfilled for $\sin^2(\xi\Delta x/2) = 0$, hence we check

  $$\sin^2(\xi\Delta x/2)((\epsilon_3+2)^2/9 - \epsilon_2^2) + (-(\epsilon_3+2)/3 + \epsilon_2^2) \le 0, \qquad \forall \sin^2(\xi\Delta x/2) \in]0,1].$$

  This is an affine expression on $\sin^2(\xi\Delta x/2)$, thus the maximum is reached on the boundary of $[0,1]$. Assume without loss of generality that $\epsilon_2 > 0$ and the standard CFL condition $\epsilon_2 \le 1$.

  - $(\epsilon_3+2)^2/9 - \epsilon_2^2 \ge 0$, corresponding to

    $$\epsilon_3 \le -2 - 3\epsilon_2, \qquad \text{or} \qquad \epsilon_3 \ge -2 + 3\epsilon_2.$$

    In this case the maximum is reached at $\sin^2(\xi\Delta x/2) = 1$, thus we want $(\epsilon_3+2)(\epsilon_3 - 1) \le 0$, hence $-2 \le \epsilon_3 \le 1$. Under the CFL condition $\epsilon_2 \le 1$ (otherwise all the computations can be adapted accordingly but no stability can be deduced), we easily find the first overall condition $-2 + 3\epsilon_2 \le \epsilon_3 \le 1$.
  - $(\epsilon_3+2)^2/9 - \epsilon_2^2 < 0$, corresponding to

    $$-2 - 3\epsilon_2 < \epsilon_3 < -2 + 3\epsilon_2.$$

    In this case the maximum is reached on $\sin^2(\xi\Delta x/2) = 0$, providing $-(\epsilon_3+2)/3 + \epsilon_2^2 \le 0$ thus comparing with the other conditions taking the CFL condition into account, we have $-2 + 3\epsilon_2^2 \le \epsilon_3 \le -2 + 3\epsilon_2$.

  Overall, the necessary and sufficient condition in this case reads $|\epsilon_2| \le 1$ and $-2 + 3\epsilon_2^2 \le \epsilon_3 \le 1$.

- Let $s_2 = 2$. In this case $\hat{\Psi}_1(\xi\Delta x, z) \equiv 0$, hence we compute

$$\frac{d\hat{\Psi}_2(\xi\Delta x, z)}{dz} = 2z + 2i\epsilon_2 \sin(\xi\Delta x),$$

  hence by the second condition in Theorem 7.7.2, we have to enforce the strict CFL condition $|\epsilon_2| < 1$.

ECOLE
DOCTORALE
DE MATHEMATIQUES
HADAMARD

Titre : Analyse numérique des schémas de Boltzmann sur réseau : des questions fondamentales aux méthodes adaptatives efficientes et précises

Mots clés : Boltzmann sur réseau, analyse numérique, multirésolution adaptative, différences finies, consistance, stabilité

Résumé : Le travail faisant l'objet de cette thèse s'inscrit dans le domaine de l'étude des méthodes numériques pour les équations aux dérivées partielles et porte une attention particulière aux schémas de Boltzmann sur réseau. Cette classe de schémas est utilisée depuis la fin des années '80, en particulier en mécanique des fluides, et se caractérise par sa grande rapidité. Cependant, les méthodes de Boltzmann sur réseau sont très gourmandes en termes d'espace mémoire et conçues pour des maillages Cartésiens uniformes. De plus, nous manquons d'outils théoriques généraux qui permettent d'en analyser la consistance, la stabilité et enfin la convergence. Le travail de thèse s'articule autour de deux axes principaux. Le premier consiste à proposer une stratégie permettant d'appliquer les méthodes de Boltzmann sur réseau à des grilles de calcul non-uniformes adaptées dynamiquement en temps, afin de réduire le coût de calcul et de stockage. Le fait de pouvoir contrôler l'erreur commise et d'être en mesure d'employer la méthode quel que soit le schéma de Boltzmann sous-jacent sont des contraintes supplémentaires à prendre en compte. Pour cela, nous proposons d'adapter dynamiquement le réseau ainsi que d'ajuster toute méthode de Boltzmann à des maillages non-uniformes en nous appuyant sur la multirésolution. Cela a permis de proposer un cadre innovant pour des maillages mobiles en respectant les contraintes posées. Ensuite, nous démontrons que la méthode proposée présente d'excellentes propriétés en termes de perturbations introduites sur le schéma originel et qu'elle permet ainsi de réduire les phénomènes parasites liés aux maillages adaptés. L'implémentation de cette procédure dans un logiciel ouvert, permettant de représenter et gérer des grilles adaptées par différentes approches dans un cadre unifié et innovant, est ensuite abordée. Le second axe de recherche consiste à donner un cadre mathématiquement rigoureux aux méthodes de Boltzmann sur réseau, lié en particulier à leur consistance vis-à-vis des EDPs visées, leur stabilité et donc leur convergence. Pour cela, nous proposons une procédure, basée sur des résultats d'algèbre, pour éliminer les moments non-conservés de n'importe quel schéma de Boltzmann sur réseau, en le transformant en un schéma aux différences finies multi-pas sur les moments conservés. Les notions de consistance et stabilité pertinentes pour les méthodes de Boltzmann sur réseau sont donc celles des schémas aux différences finies. En particulier, tous les résultats concernant ces derniers, entre autres le théorème de Lax, se transpose naturellement aux schémas de Boltzmann sur réseau. Une étape ultérieure consiste à étudier la consistance et la stabilité directement sur le schéma de départ sans devoir calculer sa méthode aux différences finies "correspondante". Cela permet d'en obtenir les équations modifiées et de montrer le bien-fondé des analyses de stabilité à la *von Neumann* couramment utilisées au sein de la communauté. Ce nouveau cadre théorique permet aussi d'étudier l'influence de l'initialisation des méthodes sur le résultat des simulations ainsi que d'entamer des études préliminaires sur la monotonie des schémas de Boltzmann sur réseau et sur leurs conditions aux limites, qui constituent des ouvertures pour des travaux futurs.

Title : Numerical analysis of lattice Boltzmann schemes: from fundamental issues to efficient and accurate adaptive methods

Keywords : lattice Boltzmann, numerical analysis, adaptive multiresolution, finite difference, consistency, stability

Abstract : The work presented in this thesis falls within the field tackling the analysis of numerical methods for Partial Differential Equations and pays particular attention to lattice Boltzmann schemes. This class of schemes has been used since the end of the 1980s, particularly in fluid mechanics, and is characterised by its great computational efficiency. However, lattice Boltzmann methods are very demanding in terms of memory space and are designed for uniform Cartesian meshes. Moreover, we lack general theoretical tools allowing us to analyse their consistency, stability and finally convergence. The work of the thesis is articulated around two main axes. The first one consists in proposing a strategy to apply lattice Boltzmann methods to non-uniform grids being adapted in time, in order to reduce the computing and storage costs. The ability to control the error and to be able to use the same approach irrespective of the underlying lattice Boltzmann scheme are additional constraints to be taken into account. To this end, we propose to dynamically adapt the lattice as well as to adjust any Boltzmann method to non-uniform meshes by relying on multiresolution analysis. This allows us to propose an innovative framework for moving meshes while respecting the posed constraints. Then, we demonstrate that the proposed method has excellent properties in terms of the perturbations of the original scheme and that it thus allows to reduce the spurious phenomena linked to the adapted meshes. The implementation of this procedure in an open-source software, allowing to represent and manage adapted grids by different approaches in a unified and innovative framework, is then addressed. The second line of research consists in giving a mathematically rigorous framework to the lattice Boltzmann methods, related in particular to their consistency with respect to the target PDEs, their stability, and thus their convergence. For this purpose, we propose a procedure, based on algebraic results, to eliminate the non-conserved moments of any lattice Boltzmann scheme, by recasting it into a multi-step Finite Difference scheme on the conserved moments. The notions of consistency and stability relevant to lattice Boltzmann methods are therefore those of Finite Difference schemes. In particular, all the results concerning the latter, among others the Lax theorem, are naturally transposed to the lattice Boltzmann schemes. A further step consists in studying the consistency and stability directly on the original scheme without having to calculate its "corresponding" Finite Difference method. This allows us to obtain the modified equations and to show the validity of the *von Neumann* stability analyses commonly used within the community. This new theoretical framework also makes it possible to study the influence of the initialization of the methods on the result of the simulations as well as to initiate preliminary studies on the monotonicity of lattice Boltzmann schemes and on their boundary conditions, which constitute openings for future work.

Institut Polytechnique de Paris
91120 Palaiseau, France