HAL open science

Schéma de relaxation pour la simulation de plasmas dans les tokamaks

Romane Hélie

To cite this version:

Romane Hélie. Schéma de relaxation pour la simulation de plasmas dans les tokamaks. Physique mathématique [math-ph]. Université de Strasbourg, 2023. Français. NNT : . tel-04034510v1

HAL Id: tel-04034510

https://theses.hal.science/tel-04034510v1

Submitted on 25 May 2023 (v1), last revised 13 Jun 2023 (v2)

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.

Thèse

INSTITUT DE RECHERCHE MATHÉMATIQUE AVANCÉE

UMR 7501

Strasbourg

présentée pour obtenir le grade de docteur de
l'Université de Strasbourg
Spécialité MATHÉMATIQUES

Romane Hélie

Schéma de relaxation pour la simulation de plasmas dans les tokamaks

Soutenue le 27 mars 2023
devant la commission d'examen

Philippe Helluy, directeur de thèse
Laurent Navoret, co-encadrant
Benjamin Graille, rapporteur
Michel Mehrenberger, rapporteur
Denise Aregba-Driollet, examinatrice
François Dubois, examinateur

https://irma.math.unistra.fr

Institut de Recherche Mathématique Avancée

Université de Strasbourg

UNIVERSITÉ DE STRASBOURG

ÉCOLE DOCTORALE 269

Mathématiques, Sciences de l'Information et de l'Ingénieur (MSII)

UMR 7501

Institut de Recherche Mathématique Avancée (IRMA)

THÈSE présentée par :

Romane HÉLIE

soutenue le **27 mars 2023**

pour obtenir le grade de : **Docteur de l'université de Strasbourg**

Discipline/ Spécialité : MATHÉMATIQUES

Schéma de relaxation pour la simulation de plasmas dans les tokamaks

THÈSE dirigée par :

M. HELLUY Philippe	Professeur, Université de Strasbourg

RAPPORTEURS :

M. GRAILLE Benjamin	Maitre de conférence, Université Paris-Saclay
M. MEHRENBERGER Michel	Professeur, Université de Marseille

AUTRES MEMBRES DU JURY :

Mme AREGBA-DRIOLLET Denise	Maitresse de conférence, Bordeaux INP
M. DUBOIS François	Professeur, Université Paris-Saclay
M. NAVORET Laurent	Maitre de conférence, Université de Strasbourg

Remerciements

Tout d'abord, je tiens à remercier mon directeur de thèse Philippe Helluy pour m'avoir fait confiance en me proposant ce sujet de recherche. Je le remercie pour tout son soutien, ses nombreuses idées, et pour son optimisme.

Je remercie également mes co-encadrants Laurent Navoret et Emmanuel Franck qui ont toujours été disponibles, impliqués et bienveillants.

Je remercie Benjamin Graille et Michel Mehrenberger d'avoir accepté de rapporter ce manuscrit, ainsi que Denise Aregba-Driollet et François Dubois pour m'avoir fait l'honneur de compléter mon jury.

Mes remerciements vont aussi à Matthieu Boileau, pour toute son aide informatique, en particulier pour la visualisation des simulations en 3 dimensions.

Je remercie également tous les membres de l'équipe Moco, pour les nombreuses conversations mathématiques échangées tout au long de cette thèse.

Merci aussi à Kévin Guillon, avec qui cela a été un plaisir de travailler durant les 6 semaines du Cemracs.

J'aimerais remercier tous les doctorants et postdoctorants croisés au cours de ces quatre années, pour les nombreux moments partagés avec eux. Je remercie en particulier mes co-bureaux du 408, les nombreux collègues qui ont partagé pendant quelques mois la bibliothèque pour l'ambiance chaleureuse (et peu studieuse) qu'ils y ont mise, et plus récemment, les collègues du 114. J'ai également une pensée pour les doctorants avec qui je suis allée en conférence, et notamment pour ceux rencontrés au Cemracs.

Enfin, pour leur soutien depuis toutes ses années, j'aimerais remercier mes grands-mères, Balou, Lola et mes parents.

Contents

1 Introduction **5**
- 1 Modélisation d'un plasma de tokamak . . . 5
- 2 Modélisation des conditions aux bords . . . 6
- 3 Difficultés numériques . . . 6
- 4 Méthodes cinétiques pour le transport . . . 7

2 Schéma de relaxation cinétique **9**
- 1 Schéma de relaxation cinétique . . . 9
  - 1.1 Système hyperbolique de loi de conservation . . . 10
  - 1.2 Représentation vectorielle cinétique . . . 10
  - 1.3 Splitting et sur-relaxation . . . 12
- 2 Vitesses cinétiques . . . 16
  - 2.1 Le modèle $D1Q2$ . . . 17
  - 2.2 Le modèle $D2Q3$ . . . 17
  - 2.3 Le modèle $D2Q4$ . . . 18
  - 2.4 Le modèle $D2Q4$ twisté . . . 19
  - 2.5 Le modèle $D2Q5$ . . . 19
- 3 Conclusion . . . 20

3 Equivalent system in the general case **21**
- 1 Computation of the equivalent system . . . 22
- 2 Applications . . . 24
  - 2.1 Equivalent system of the $D1Q2$ model . . . 25
  - 2.2 Equivalent system of the $D2Q3$ model . . . 25
  - 2.3 Equivalent system of the $D2Q4$ model . . . 25
  - 2.4 Equivalent system of the $D2Q4$ twisted model . . . 26
- 3 Numerical validation of the equivalent systems . . . 27
  - 3.1 Validation of the equivalent system of the $D2Q3$ model . . . 27
  - 3.2 Validation of the equivalent system of the $D2Q4$ model . . . 27
- 4 Conclusion . . . 29

4 Application to the transport equation **31**
- 1 Equivalent system on (w, Y) . . . 32
- 2 Equivalent equation on w . . . 32
- 3 Equivalent equations of the $D1Q2$ with the operator $\mathcal{M}$ . . . 33

4 Applications for the symmetric operator $\mathcal{S}$ . . . 33
4.1 For the $D1Q2$ model . . . 34
4.2 For the $D2Q3$ model . . . 36
4.3 For the $D2Q4$ model . . . 37
5 Comparison of the equivalent system on (w, Y) and the equivalent equation on w for the $D1Q2$ model . . . 38
5.1 Particular solution of the equivalent equation . . . 38
5.2 Particular solution of the equivalent system . . . 39
5.3 Numerical comparison of w . . . 40
5.4 Numerical comparison of y . . . 41
6 Conclusion . . . 43

5 Two-scale convergence **45**
1 Two-scale convergence . . . 45
2 Conclusion . . . 50

6 Numerical stability **51**
1 Entropy stability of the kinetic model . . . 51
1.1 Dual kinetic entropy representation . . . 51
1.2 Reverse construction . . . 55
1.3 Application to the transport equation . . . 55
1.4 Application to a truly nonlinear system . . . 57
2 Stability conditions . . . 60
2.1 Hyperbolicity condition . . . 60
2.2 The $D1Q2$ model . . . 61
2.3 The $D2Q3$ model . . . 63
2.4 The $D2Q4$ model . . . 65
3 Boundary conditions . . . 70
3.1 The $D1Q2$ model . . . 70
3.2 The $D2Q4$ model . . . 79

7 Kinetic over-relaxation method for the convection equation with Fourier solver **95**
1 Introduction . . . 95
2 Kinetic over-relaxation approximation of the convection equation . . . 95
3 Numerical results . . . 98
4 Conclusion . . . 101

8 Parallel kinetic scheme for transport equations in complex toroidal geometry **103**
1 Introduction . . . 103
2 Transport equation and reformulation . . . 105
2.1 In cylindrical coordinates . . . 105
2.2 Kinetic relaxation method . . . 106
3 Transport solver . . . 108
3.1 Transport in the toroidal direction: LBM . . . 108
3.2 Transport in the poloidal direction: implicit DG . . . 108

3.3 Parallelization . . . 111
4 Kinetic solver . . . 111
4.1 Splitting . . . 111
4.2 CFL condition . . . 112
4.3 Boundary conditions . . . 112
4.4 Validation in two-dimensional geometry . . . 113
4.5 Validation in a 3D periodic cylinder . . . 115
5 Applications to plasma dynamics . . . 116
5.1 Drift-kinetic like model . . . 117
5.2 Two-dimensional diocotron test-case . . . 118
5.3 Diocotron test-case in a periodic cylinder . . . 120
5.4 Parallel efficiency: weak scaling . . . 124
6 Conclusion . . . 125

9 Optimization of spatial control strategies for population replacement, application to *Wolbachia* 127
1 Introduction and state of the art . . . 127
2 Modelling . . . 130
2.1 Model with two compartments . . . 130
2.2 Reduction for large fecundity . . . 131
2.3 Toward an optimal control problem . . . 134
3 Main results . . . 136
4 Proofs . . . 137
4.1 Model reduction . . . 137
4.2 Analysis of the optimal control problem ($\mathcal{P}_{\text{reduced}}$) . . . 143
5 Numerical experiments . . . 153
6 Perspectives . . . 157

10 Conclusion 159

Annexes 161
A Calcul des équations équivalentes . . . 161
A.1 Linéarisation . . . 163
A.2 Applications . . . 166
B Code Maple . . . 169
B.1 Définition des différents modèles cinétiques . . . 169
B.2 Définition des opérateurs et autres fonctions . . . 170
B.3 Système équivalent en (w, Y) . . . 173
B.4 Simplification du système équivalent en (w, Y) en une équation équivalente en w . . . 173
B.5 Condition d'hyperbolicité . . . 174
B.6 Matrice d'entropie . . . 175

Bibliography 176

Chapitre 1

Introduction

1 Modélisation d'un plasma de tokamak

Cette thèse s'inscrit en partie dans un projet de longue haleine qui consiste à construire un tokamak afin de produire de l'énergie grâce à la fusion thermonucléaire contrôlée. Cette source d'énergie est inépuisable et complètement décarbonée. Un tokamak est un dispositif, inventé dans les années 50, constitué d'une chambre à vide torique, dans laquelle est confiné un plasma d'hydrogène grâce à un fort champ magnétique. De nombreux tokamaks existent dans le monde. Depuis quelques années, un projet international, le projet de tokamak ITER (International Thermonuclear Experimental Reactor), consiste à construire à Cadarache dans le sud de la France un tokamak de 6 mètres de diamètre, qui sera le plus grand jamais construit. ITER devrait produire son premier plasma d'ici 2030. ITER est un projet expérimental. Son successeur, DEMO est prévu pour produire réellement de l'énergie.

Pour des raisons d'ingénierie, il est nécessaire de disposer de modèles numériques pour simuler sur ordinateur le fonctionnement des tokamaks. Le modèle mathématique fondamental pour les plasmas est constitué des équations de Vlasov-Poisson. L'équation de Vlasov est une équation de transport en temps posée dans un espace à 6 dimensions (espace des positions et des vitesses) qui décrit l'évolution de la distribution des particules chargées. La force qui accélère les particules contient deux composantes : une composante magnétique, imposée par les bobines du tokamak et une composante électrique auto-consistante. Le champ électrique est solution de l'équation de Poisson qui décrit l'évolution du potentiel électrique en fonction de la charge. L'équation de Poisson est posée dans un espace à trois dimensions. La charge électrique est obtenue en intégrant en vitesses la fonction de distribution des particules. Finalement, le système de Vlasov-Poisson est un système de deux équations linéaires couplées, mais le couplage est non linéaire. La géométrie du tokamak est relativement simple, mais c'est surtout la géométrie du champ magnétique imposée qui fixe les trajectoires des particules. Il est donc important de résoudre les équations sur des maillages adaptés à cette géométrie, ce qui conduit à des schémas numériques finalement assez complexes. Comme le système de Vlasov-Poisson est posé dans un espace à 6 dimensions, il est actuellement impossible de le résoudre numériquement, même sur un supercalculateur. Par conséquent, des modèles simplifiés ou modèles réduits ont été développés afin de rendre les simulations faisables. Le modèle de choix, qui nécessite encore une puissance de calcul conséquente, est

le modèle gyrocinétique qui permet de passer de 6 à 5 dimensions [51]. D'autres réductions conduisent à des modèles encore plus simples, comme le modèle dit de drift que nous étudierons dans cette thèse. La propriété commune des modèles réduits que nous étudierons est que ce sont des équations de transport conservatives, dont la vitesse de transport est calculée à partir de la solution d'une équation de Poisson. L'équation de transport conservative est un cas particulier de système de lois de conservation hyperbolique. L'objectif de cette thèse est donc d'étudier une nouvelle méthode numérique pour la résolution des systèmes de lois de conservation hyperboliques. Cette méthode sera particulièrement optimisée pour être appliquée à des équations de transport à vitesse variable pour la simulation des plasmas de tokamak.

2 Modélisation des conditions aux bords

Le traitement des conditions aux limites est un aspect particulièrement important dès qu'il s'agit de réaliser des simulations dans des conditions réalistes. Pour un système hyperbolique quelconque, le nombre de conditions à imposer en un point du bord du domaine de calcul dépend en général de l'évolution de la solution. La façon d'imposer la condition est également importante pour s'assurer de la stabilité et de la précision de la solution numérique. Un schéma parfaitement stable et d'ordre très élevé dans le domaine de calcul peut facilement retomber à l'ordre 1 ou devenir instable avec des conditions aux limites mal conçues. Les conditions admissibles pour les équations de transport sont relativement simples à comprendre : il ne faut les imposer que sur les bords du domaine de calcul où la vitesse est rentrante. Mais si la théorie mathématique est simple à comprendre, la pratique numérique s'avère plus complexe.

3 Difficultés numériques

Les problèmes de transport issus de la physique des plasmas de tokamak présentent certaines caractéristiques particulières. D'une part, la taille des maillages et l'évolution rapide des champs interdisent en pratique l'utilisation de schémas implicites pour l'équation de transport. L'équation de Poisson nécessite la résolution d'un système linéaire à chaque pas de temps, mais la matrice de ce système est fixée et, en général, liée à un problème bidimensionnel dans les plans poloidaux du tokamak. Les schémas explicites imposent un pas de temps contraint par la condition de stabilité de Courant-Friedrichs-Lewy (CFL). En pratique, cette contrainte CFL, purement numérique, peut imposer des pas de temps trop petits par rapport à la précision requise, ce qui rend les calculs très lents. Ce problème est particulièrement contraignant pour les particules rapides, correspondant aux queues de distribution, donc peu nombreuses, qui imposent le pas de temps à toute la simulation. D'autre part, les solutions calculées sont souvent turbulentes et présentent donc de fortes variations qui imposent des maillages très fins. Il est important de maitriser aussi le signe de la fonction de distribution, qui doit rester positive.

4 Méthodes cinétiques pour le transport

Dans cette thèse, nous allons développer une nouvelle méthode cinétique pour résoudre les systèmes hyperboliques de lois de conservation et plus particulièrement l'équation de transport à vitesse variable qui apparait dans les modèles de drift pour les plasmas de tokamak. La caractéristique principale de cette méthode est qu'elle permet de construire des schémas explicites sans CFL. Nous présenterons cette méthode, en proposerons une analyse de stabilité, y compris en présence de conditions aux limites. Enfin, nous appliquerons cette méthode pour des modèles de drift issus de la physique des plasmas de tokamak.

Le plan du document est le suivant :

- Ce chapitre (Chapitre 1) constitue l'introduction de la thèse.
- Le Chapitre 2 est consacré à la description de la méthode cinétique pour représenter un système de lois de conservation. Dans [13], Bouchut introduit des modèles cinétiques dont les inconnues sont des fonctions de distribution vectorielles. Ces fonctions de distributions sont solutions d'un système d'équations de transport à vitesses constantes, couplées par des termes de retour vers des distributions d'équilibre. Avec un bon choix de vitesses cinétiques et des fonctions d'équilibre, le modèle cinétique peut formellement approcher n'importe quel système de lois de conservation hyperbolique.
  Le modèle cinétique formel contient un terme raide de retour vers l'équilibre. Ce terme couple toutes les équations de transport et rend la résolution numérique difficile. Afin de simplifier la résolution, il est classique d'intégrer en temps le modèle par un schéma de *splitting* (ou de décomposition) où, sur un pas de temps Δt, nous séparons les étapes de transport libres et de relaxation locales. Nous introduisons alors le paramètre de relaxation ω tel que $1 \leqslant \omega \leqslant 2$. Le choix $\omega = 1$ correspond à un retour exact à l'équilibre à la fin de chaque transport libre (projection). Le choix $\omega = 2$, dit de sur-relaxation, est intéressant, car dans ce cas, le schéma de splitting est d'ordre 2 en temps. Dans la suite du chapitre, nous introduisons quelques modèles cinétiques concrets qui seront utilisés au cours de la thèse.
- Dans le Chapitre 3, nous utilisons la méthode de l'équation équivalente pour prouver la validité de la construction cinétique. Nous nous limitons au cas de la sur-relaxation $\omega = 2$. En effet, avec ce choix, des termes raides disparaissent dans les développements de Taylor en Δt. Nous obtenons ainsi les équations équivalentes des modèles concrets du chapitre 2 dans le cas de l'équation de transport à vitesse variable. Nous validons ensuite quantitativement cette analyse sur des cas-tests simples en 2 dimensions.
- Dans le Chapitre 4, nous développons l'analyse nécessaire dans le cas $1 \leqslant \omega < 2$. Dans ce cas, l'équation équivalente contient des termes raides en $1/\Delta t$. Ce terme raide exprime le retour rapide vers 0 de la variable d'écart à l'équilibre. Il est possible, grâce à une analyse de Chapman-Enskog, de supprimer la variable d'écart et d'obtenir une équation équivalente uniquement sur la variable conservative. C'est généralement cette équation qui est directement écrite dans l'analyse habituelle (voir [34, 24], par exemple).
- Dans le Chapitre 5, nous proposons une analyse alternative afin de modéliser l'effet des termes raides de l'équation équivalente au moyen d'une analyse double échelle. Cette

analyse consiste à introduire deux variables de temps pour capturer l'échelle rapide et l'échelle lente. La solution en temps est considérée comme une fonction de deux variables temporelles. Lorsque $\Delta t \to 0$, nous montrons que la solution tend au sens de la limite double échelle vers une fonction à trois variables : le temps rapide, le temps lent et l'espace. Il est possible d'identifier l'EDP à trois variables satisfaite par cette limite. Cette analyse permet de mieux comprendre ce qui se passe dans la couche limite en temps pour des conditions initiales loin de l'équilibre.

- Dans le Chapitre 6, nous revenons à une analyse plus classique afin d'établir des résultats de stabilité du schéma cinétique avec splitting en temps. Nous rappelons d'abord la théorie de stabilité entropique du schéma cinétique vectoriel, établie par Bouchut [13] et étendue par Dubois [35]. Nous montrons que cette théorie s'étend au schéma avec sur-relaxation et aux systèmes hyperboliques non linéaires. Cette analyse repose sur un calcul d'entropies cinétiques duales. Ce calcul est en général assez compliqué. Mais il devient beaucoup plus simple si nous nous limitons à l'équation de transport conservative, qui est linéaire. Cette analyse permet d'analyser également la stabilité des conditions aux limites. Nous terminons donc le chapitre avec la présentation d'une technique qui permet d'appliquer des conditions aux limites naturelles et stables avec le schéma cinétique vectoriel.

- Nous continuons ensuite avec le Chapitre 7, qui est le premier chapitre d'application numérique à des calculs d'instabilités en physique des plasmas. Dans ce travail (issu d'un article présenté à la conférence FVCA 9), nous construisons un schéma cinétique d'ordre 4 en temps et en espace pour un modèle de centre-guide couplé à une équation de Poisson. Pour cela, nous résolvons l'étape de transport par une transformée de Fourier discrète et nous utilisons une technique de montée en ordre palindromique [71]. Ce schéma nous permet de calculer avec une grande précision une instabilité de Kelvin-Helmholtz.

- Dans le Chapitre 8, nous présentons des résultats tridimensionnels, obtenus avec le schéma cinétique sur un modèle de drift. Le code tridimensionnel utilisé est parallélisé de façon efficace grâce aux bibliothèques MPI et OpenMP. L'originalité est d'utiliser des approximations différentes dans les directions toroïdales et poloïdales. Dans les plans poloïdaux, les équations cinétiques sont résolues avec des approximations de Galerkin-Discontinu (GD) sans CFL. Cette approximation repose sur un schéma explicite basé sur un graphe des tâches acyclique. Dans la direction toroïdale, les équations cinétiques sont résolues avec une technique de Lattice-Boltzmann. En pratique, chaque plan poloïdal est associé à un unique noeud MPI et les décalages sont simplement des échanges MPI entre plans poloïdaux, ce qui rend l'algorithme particulièrement simple. Ce code optimisé nous permet de calculer des instabilités de type diocotron.

- Le Chapitre 9 conclut la thèse. Nous y présentons un travail sur la modélisation du contrôle d'une population de moustiques par des lâchers de moustiques infectés par la bactérie *Wolbachia*. Ce travail est complètement indépendant du reste du manuscrit.

Chapitre 2

Schéma de relaxation cinétique

1 Schéma de relaxation cinétique

Comme expliqué dans l'introduction, la résolution numérique des systèmes de lois de conservation issus de la physique des plasmas conduit à diverses difficultés. Parmi ces difficultés, nous pouvons noter des problèmes de précision des schémas lorsqu'il y a de la turbulence ou de forts gradients. Ensuite, il y a des problèmes de coût de calcul liés à la contrainte de stabilité de type CFL pour les schémas explicites. Enfin, un aspect important est de pouvoir prendre en compte de façon stable et précise les conditions aux limites du modèle. Cette thèse est consacrée à l'étude d'une famille de schémas numériques, les schémas cinétiques, qui permettent de construire des schémas simples, précis et robustes.

Dans ce chapitre, nous décrivons comment représenter un système de lois de conservation hyperbolique par un système cinétique avec un nombre fini de vitesses de transport. Ce type de représentation imite la théorie cinétique des gaz de Boltzmann, bien qu'il s'agisse d'une représentation purement abstraite, qui n'a en général pas de sens physique.

L'idée d'utiliser une représentation cinétique pour construire des schémas numériques n'est pas nouvelle. Le premier travail utilisant cette idée est sans doute dû à Deshpande [29]. Il a proposé dès les années 1980 des schémas de volumes finis pour résoudre les équations d'Euler compressibles. Le flux numérique de ces schémas est calculé à partir des distributions Maxwelliennes en vitesse. Le modèle cinétique possède donc une infinité de vitesses. Il n'est pas résolu directement, mais exploité comme intermédiaire pour la construction du schéma cinétique. D'autres auteurs ont étendu cette idée. Voir par exemple [77, 16, 25].

Les schémas de type *Lattice-Boltzmann Method* (LBM) reposent sur une représentation cinétique avec un nombre fini de vitesses. Comme le nombre de vitesses est fini, il est possible de résoudre directement les équations de transport, sans intégrer en vitesse. Cette approche est très intéressante, car elle permet, entre autres, de construire des schémas très performants, explicites, mais sans contrainte de stabilité sur le pas de temps.

Les premiers schémas LBM étaient présentés comme des approximations de la théorie cinétique de Boltzmann [20, 91]. Ils permettent de résoudre les équations de Navier-Stokes faiblement compressibles. Ils reposent sur une fonction de distribution scalaire. L'approche

cinétique a ensuite été généralisée à n'importe quel système hyperbolique par Bouchut [13] ou Aregba-Natalini [5, 6]. L'utilisation d'une fonction de distribution vectorielle permet de représenter aussi les écoulements fortement compressibles.

Nous rappelons dans ce chapitre la théorie de Bouchut dans le cas général d'un système de lois de conservation hyperbolique. L'objectif est de décrire les différentes représentations cinétiques : $D1Q2$, $D2Q3$, $D2Q4$ et $D2Q5$ qui seront utilisées dans la suite de la thèse et qui seront particularisées à l'équation de transport à vitesse variable.

1.1 Système hyperbolique de loi de conservation

Nous nous intéressons à la résolution numérique d'un système de lois de conservation hyperbolique. Le vecteur d'inconnues est noté $\boldsymbol{w}(\boldsymbol{x}, t) \in \mathbb{R}^r$. Il dépend du temps t et du vecteur d'espace $\boldsymbol{x} = (x^1, \ldots, x^d) \in \mathbb{R}^d$, où d est la dimension de l'espace (en pratique, $d = 1, 2$ ou 3). Nous considérons un système de la forme

$$\partial_t \boldsymbol{w} + \nabla \cdot \boldsymbol{q}(\boldsymbol{w}) = 0, \tag{2.1}$$

où $\boldsymbol{q}$ est le flux, avec $q^i : \mathbb{R}^r \to \mathbb{R}^r$ pour $i = 1, \ldots, d$. Le système est supposé être hyperbolique : la jacobienne du flux par rapport à $\boldsymbol{w}$ est supposée être une matrice diagonalisable, avec des valeurs propres réelles.

1.2 Représentation vectorielle cinétique

Afin de résoudre numériquement (2.1), nous souhaitons construire une représentation cinétique de ce système hyperbolique.

Pour les systèmes à une dimension, c'est-à-dire $d = 1$, et 2 vitesses cinétiques, nous verrons que cette construction revient à la relaxation de Jin-Xin [61].

Le cas général où $d \geqslant 1$ est étudié dans [13, 5, 6, 22].

Nous considérons un ensemble de n_v vecteurs de $\mathbb{R}^d$: $\Lambda = \{\boldsymbol{\lambda}_k = (\lambda_k^1, \ldots, \lambda_k^d)^T \in \mathbb{R}^d, k = 1, \ldots, n_v\}$. Les vecteurs $\boldsymbol{\lambda}_k$ sont appelés vitesses cinétiques. Nous supposons que le rang de Λ est maximal, c'est-à-dire qu'il est égal à d. Cela impose donc que $n_v \geqslant d$. À chaque vitesse cinétique $\boldsymbol{\lambda}_k$, nous associons une inconnue cinétique $\boldsymbol{f}_k(\boldsymbol{x}, t) \in \mathbb{R}^r$. Les inconnues cinétiques et la donnée macroscopique $\boldsymbol{w}$ sont liées par

$$\boldsymbol{w} = \sum_{k=1}^{n_v} \boldsymbol{f}_k. \tag{2.2}$$

Nous considérons ensuite le système d'équations cinétiques

$$\partial_t \boldsymbol{f}_k + \boldsymbol{\lambda}_k \cdot \nabla \boldsymbol{f}_k = \frac{1}{\varepsilon}(\boldsymbol{f}_k^{eq}(\boldsymbol{w}) - \boldsymbol{f}_k), \qquad \forall k = 1, \ldots, n_v. \tag{2.3}$$

Dans ce système d'équations cinétiques, interviennent des fonctions cinétiques à l'équilibre, $\boldsymbol{f}_k^{eq}$ sur lesquelles nous allons imposer des contraintes pour que le système cinétique soit

consistant avec le système de lois de conservation hyperbolique (2.1). En sommant les équations cinétiques (2.3) sur k, nous obtenons

$$\partial_t \boldsymbol{w} + \sum_{k=1}^{n_v} (\boldsymbol{\lambda}_k \cdot \nabla \boldsymbol{f}_k) = \frac{1}{\varepsilon} \left(-\boldsymbol{w} + \sum_{k=1}^{n_v} \boldsymbol{f}_k^{eq}(\boldsymbol{w}) \right). \tag{2.4}$$

Formellement, lorsque $\varepsilon \to 0$, nous avons $\boldsymbol{f}_k \simeq \boldsymbol{f}_k^{eq}(\boldsymbol{w})$. Si nous supposons que

$$\sum_{k=0}^{n_v} \boldsymbol{f}_k^{eq}(\boldsymbol{w}) = \boldsymbol{w},$$

alors, lorsque $\varepsilon \to 0$, (2.4) devient

$$\partial_t \boldsymbol{w} + \nabla \cdot \left(\sum_{k=1}^{n_v} \boldsymbol{\lambda}_k \boldsymbol{f}_k^{eq}(\boldsymbol{w}) \right) = 0. \tag{2.5}$$

En conclusion, lorsque $\varepsilon \to 0$, le système cinétique (2.3) est consistant avec le système hyperbolique initial (2.1) à condition que

$$\sum_{k=1}^{n_v} \boldsymbol{f}_k^{eq}(\boldsymbol{w}) = \boldsymbol{w} \quad \text{et} \quad \boldsymbol{q}^i(\boldsymbol{w}) = \sum_{k=1}^{n_v} \lambda_k^i \boldsymbol{f}_k^{eq}(\boldsymbol{w}). \tag{2.6}$$

Nous obtenons donc $r(d+1)$ relations de consistance vérifiées par rn_v inconnues (les composantes des n_v vecteurs cinétiques d'équilibre $\boldsymbol{f}_k^{eq}(\boldsymbol{w})$). Le nombre de vitesses cinétiques n_v doit donc nécessairement être supérieur ou égal à $d+1$. Ces conditions de consistance peuvent également être écrites sous forme matricielle

$$\boldsymbol{Q}(\boldsymbol{w}) = V \boldsymbol{F}^{eq}(\boldsymbol{w}),$$

avec $\boldsymbol{F}^{eq}(\boldsymbol{w}) = \begin{pmatrix} \boldsymbol{f}_1^{eq}(\boldsymbol{w}) \\ \vdots \\ \boldsymbol{f}_{n_v}^{eq}(\boldsymbol{w}) \end{pmatrix}$, $\boldsymbol{Q}(\boldsymbol{w}) = \begin{pmatrix} \boldsymbol{z}_0^{eq}(\boldsymbol{w}) \\ \vdots \\ \boldsymbol{z}_{n_v-1}^{eq}(\boldsymbol{w}) \end{pmatrix}$ et V une matrice inversible, carrée d'ordre n_v.

Les coefficients de la matrice V sont donnés par $v_{i,j} = m^{i-1}(\boldsymbol{\lambda}_j)$, où les fonctions moments $m^i \in \mathbb{R}[X_1, \cdots, X_d]$ sont des polynômes sur les composantes des vitesses cinétiques.

La première relation de consistance (2.6) nous donne

$$m^0(\boldsymbol{\lambda}) = 1 \quad \text{et} \quad \boldsymbol{z}_0^{eq}(\boldsymbol{w}) = \boldsymbol{w},$$

tandis que la seconde impose

$$m^i(\boldsymbol{\lambda}) = \lambda^i \quad \text{et} \quad \boldsymbol{z}_i^{eq}(\boldsymbol{w}) = \boldsymbol{q}^i(\boldsymbol{w}), \quad \text{pour tout } i = 1, \cdots, d.$$

Lorsque $n_v > d+1$, nous disposons de $n_v - d - 1$ degrés de liberté. Nous pouvons donc choisir les moments m^i et leur équilibre associé $\boldsymbol{z}_i^{eq}(\boldsymbol{w})$, pour tout $i = d+1, \ldots, n_v - 1$, de manière à ce que la matrice V soit inversible.

Finalement, nous avons la matrice $\boldsymbol{Q}(\boldsymbol{w})$ et la matrice des moments V de la forme

$$\boldsymbol{Q}(\boldsymbol{w}) = \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{q}^1(\boldsymbol{w}) \\ \vdots \\ \boldsymbol{q}^d(\boldsymbol{w}) \\ \boldsymbol{z}^{eq}_{d+1}(\boldsymbol{w}) \\ \vdots \\ \boldsymbol{z}^{eq}_{n_v-1}(\boldsymbol{w}) \end{pmatrix} \quad \text{et} \quad V = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \lambda^1_1 & \lambda^1_2 & \cdots & \lambda^1_{n_v} \\ \vdots & \vdots & & \vdots \\ \lambda^d_1 & \lambda^d_2 & \cdots & \lambda^d_{n_v} \\ m^{d+1}_1 & m^{d+1}_2 & \cdots & m^{d+1}_{n_v} \\ \vdots & \vdots & & \vdots \\ m^{n_v-1}_1 & m^{n_v-1}_2 & \cdots & m^{n_v-1}_{n_v} \end{pmatrix}. \tag{2.7}$$

En inversant la matrice V, nous obtenons les vecteurs cinétiques d'équilibre $\boldsymbol{f}^{eq}(\boldsymbol{w})$ à partir de la donnée macroscopique $\boldsymbol{w}$ et du flux $\boldsymbol{q}$

$$\boldsymbol{F}^{eq}(\boldsymbol{w}) = V^{-1}\boldsymbol{Q}(\boldsymbol{w}). \tag{2.8}$$

1.3 Splitting et sur-relaxation

Dans l'équation (2.3), le terme source BGK

$$\frac{1}{\varepsilon}(\boldsymbol{f}^{eq}_k(\boldsymbol{w}) - \boldsymbol{f}_k)$$

a un intérêt théorique, mais en pratique, il couple de manière non linéaire les équations cinétiques. Il est difficile de résoudre directement les équations de transport couplées au terme de relaxation. Nous allons donc approcher le système cinétique (2.3) par une méthode de *splitting*. Nous supposons que les fonctions cinétiques $\boldsymbol{f}_k(\boldsymbol{x}, t)$ sont connues au temps t pour tout $\boldsymbol{x}$. Nous rassemblons toutes les inconnues cinétiques $\boldsymbol{f}_k$ dans le vecteur

$$\boldsymbol{F} = \begin{pmatrix} \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_{n_v} \end{pmatrix}. \tag{2.9}$$

Nous construisons un opérateur non linéaire, que nous appellerons l'opérateur de Lattice-Boltzmann (LBO : *Lattice Boltzmann Operator*), $\mathcal{M}^f(\Delta t)$ qui prend les données cinétiques $\boldsymbol{F}(\cdot, t)$ au temps t et renvoie les données cinétiques au temps $t + \Delta t$

$$\boldsymbol{F}(\cdot, t + \Delta t) = \mathcal{M}^f(\Delta t)\boldsymbol{F}(\cdot, t).$$

L'opérateur LBO est le résultat d'un algorithme de splitting dans lequel à chaque pas de temps Δt, les variables cinétiques $\boldsymbol{f}_k$ vérifient les équations de transport à vitesses constantes $\boldsymbol{\lambda}_k$, puis sont relaxées vers l'état d'équilibre.

Étape de transport

Premièrement, nous résolvons les équations de transport au pas de temps Δt, pour tout $k = 1, \ldots, n_v$,

$$\partial_t \boldsymbol{g}_k + \boldsymbol{\lambda}_k \cdot \nabla \boldsymbol{g}_k = 0, \tag{2.10}$$

avec la condition initiale

$$\boldsymbol{g}_k(\cdot, 0) = \boldsymbol{f}_k(\cdot, t).$$

Cela nous permet de calculer la variable conservative

$$\boldsymbol{w}(\cdot, t + \Delta t) = \sum_{i=0}^{d} \boldsymbol{g}_i(\cdot, \Delta t).$$

Nous notons $\mathcal{T}^f(\Delta t)$ l'opérateur associé à cette étape de transport

$$\mathcal{T}^f(\Delta t)(\boldsymbol{F}(., t)(\boldsymbol{x})) = \boldsymbol{G}(\boldsymbol{x}, \Delta t), \tag{2.11}$$

où $\boldsymbol{G} = \begin{pmatrix} \boldsymbol{g}_1 & \cdots & \boldsymbol{g}_{n_v} \end{pmatrix}^T$.

Les équations de transport à vitesse constante peuvent être résolues avec diverses méthodes. Dans cette thèse, nous testerons trois approches différentes :

- Dans les chapitres 3, 4, 5 et 6, les équations de transport seront résolues de manière exacte par une méthode de Lattice-Boltzmann.
- Dans le chapitre 7, nous utiliserons une méthode de Fourier.
- Dans le chapitre 8, nous exploiterons la géométrie cylindrique du tokamak. Cela nous permettra de distinguer les vitesses cinétiques des plans poloïdaux, pour lesquelles nous utiliserons une méthode de Galerkin Discontinu, des deux vitesses de la direction toroïdale, pour lesquelles nous emploierons une méthode des caractéristiques, de type Lattice-Boltzmann.

Étape de relaxation

Après l'étape de transport, nous appliquons la formule de relaxation

$$\boldsymbol{f}_k(\cdot, t + \Delta t) = \omega \boldsymbol{f}_k^{eq}(\boldsymbol{w}(\cdot, t + \Delta t)) + (1 - \omega)\boldsymbol{g}_k(\cdot, \Delta t), \tag{2.12}$$

où $\omega \in [1, 2]$ est un paramètre de relaxation.

Si ce paramètre $\omega = 1$, nous retrouvons l'algorithme de splitting classique d'ordre 1 de Jin-Xin, où $\boldsymbol{f}_k = \boldsymbol{f}_k^{eq}(\boldsymbol{w})$, c'est-à-dire que les inconnues cinétiques sont projetées sur l'état d'équilibre à la fin de chaque pas de temps. La sur-relaxation correspond à $\omega = 2$:

$$\boldsymbol{f}_k(\boldsymbol{x}, t + \Delta t) = 2\boldsymbol{f}_k^{eq}(\boldsymbol{w}(\boldsymbol{x}, t + \Delta t)) - \boldsymbol{g}_k(\boldsymbol{x}, \Delta t). \tag{2.13}$$

Nous notons $\mathcal{R}_\omega^f$ l'opérateur associé à cette étape de relaxation

$$\mathcal{R}_\omega^f(\boldsymbol{G}(., \Delta t)(\boldsymbol{x})) = \omega \boldsymbol{F}^{eq}(\mathbb{1} \cdot \boldsymbol{G}(\boldsymbol{x}, \Delta t)) + (1 - \omega)\boldsymbol{G}(\boldsymbol{x}, \Delta t), \tag{2.14}$$

où $\mathbb{1} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix} \in \mathbb{R}^{n_v}$. En effet :

$$\boldsymbol{w}(\boldsymbol{x}, \Delta t) = \mathbb{1} \cdot \boldsymbol{G}(\boldsymbol{x}, \Delta t) = \sum_{k=0}^{d} \boldsymbol{g}_k(\boldsymbol{x}, \Delta t).$$

Composition

Il est bien connu (voir par exemple les travaux de Strang [88]) que l'enchaînement des étapes de compositions est essentiel pour obtenir une méthode de splitting d'ordre élevé. Une synthèse des schémas de composition pour approcher des systèmes d'équations différentielles est donnée dans [71]. Par ailleurs si l'équation à approcher contient des termes raides, ce qui est le cas de (2.3), l'approche du splitting classique de Strang ne permet pas d'obtenir l'ordre deux en temps [61].

Par la suite, nous noterons en majuscule calligraphique les opérateurs de paramètre Δt d'un espace fonctionnel E dans E. Par exemple, on pourra considérer $E = \left(L^2(\mathbb{R}^d)\right)^{n_v}$ ou $E = \left(\mathcal{C}^1(\mathbb{R}^d)\right)^{n_v}$.

Nous pouvons alors définir l'opérateur $\mathcal{M}^f$ associé à la succession d'une étape de transport puis d'une étape de relaxation

$$\boldsymbol{F}(\boldsymbol{x}, t + \Delta t) = \mathcal{M}^f(\Delta t)(\boldsymbol{F}(\boldsymbol{x}, t)).$$

Cet opérateur s'écrit

$$\mathcal{M}^f(\Delta t) = \mathcal{R}^f_\omega \circ \mathcal{T}^f(\Delta t), \tag{2.15}$$

avec $\mathcal{T}^f$ l'opérateur de transport défini par (2.11) et $\mathcal{R}^f_\omega$ l'opérateur de relaxation défini par (2.14).

Nous pouvons également composer ces opérateurs de transport $\mathcal{T}^f$ et de relaxation $\mathcal{R}^f_\omega$ afin de créer l'opérateur

$$\mathcal{H}^f(\Delta t) = \mathcal{T}^f\left(\frac{\Delta t}{2}\right) \circ \mathcal{R}^f_\omega \circ \mathcal{T}^f\left(\frac{\Delta t}{2}\right), \tag{2.16}$$

ou l'opérateur

$$\begin{aligned} \mathcal{S}^f(\Delta t) &= \mathcal{H}^f\left(\frac{\Delta t}{2}\right) \circ \mathcal{H}^f\left(\frac{\Delta t}{2}\right), \\ &= \mathcal{T}^f\left(\frac{\Delta t}{4}\right) \circ \mathcal{R}^f_\omega \circ \mathcal{T}^f\left(\frac{\Delta t}{2}\right) \circ \mathcal{R}^f_\omega \circ \mathcal{T}^f\left(\frac{\Delta t}{4}\right). \end{aligned} \tag{2.17}$$

Ces trois opérateurs ont des propriétés différentes pour la précision.

Définition 1.1. *On dit qu'une famille d'opérateurs de E dans E dépendant d'un paramètre Δt*

$$\begin{array}{rcccc} \psi(\Delta t) & : & E & \to & E \\ & & \begin{pmatrix} w(\cdot) \\ \boldsymbol{Y}(\cdot) \end{pmatrix} & \mapsto & \psi(\Delta t)\begin{pmatrix} w(\cdot) \\ \boldsymbol{Y}(\cdot) \end{pmatrix} \end{array}$$

est symétrique en temps si

- $\psi(-\Delta t) = \psi(\Delta t)^{-1}$,
- $\psi(0) = Id$, *où* Id *est l'opérateur identité, tel que* $Id(\boldsymbol{f}) = \boldsymbol{f}$.

Proposition 1.1. *Dans le cas de la sur-relaxation, c'est-à-dire lorsque $\omega = 2$, l'opérateur $\mathcal{S}^f$ est symétrique en temps.*

Démonstration. *Vérifions les deux conditions.*

- *Comme* $\mathcal{T}^f(-\Delta t) \circ \mathcal{T}^f(\Delta t) = Id$ *et* $\mathcal{R}_2^f \circ \mathcal{R}_2^f = Id$, *on a* $\mathcal{S}^f(-\Delta t) \circ \mathcal{S}^f(\Delta t) = Id$.
- *Comme* $\mathcal{T}^f(0) = Id$ *et* $\mathcal{R}_2^f \circ \mathcal{R}_2^f = Id$, *on en déduit que* $\mathcal{S}^f(0) = Id$.

$\mathcal{S}^f$ *est donc bien un opérateur symétrique en temps.*

Remarque 1.1. *Dans la suite, nous appellerons* $\mathcal{S}^f$ *"l'opérateur symétrique", même lorsque* $\omega \neq 2$.

Remarque 1.2. *L'opérateur* $\mathcal{H}^f$ *n'est pas symétrique en temps. En effet,* $\mathcal{H}^f(0) = \mathcal{R}_\omega^f \neq Id$, *la seconde condition de symétrie n'est donc pas respectée. De même, l'opérateur* $\mathcal{M}^f$ *n'est pas symétrique en temps car* $\mathcal{M}^f(0) = \mathcal{R}_\omega^f \neq Id$ *et* $\mathcal{M}^f(-\Delta t) \neq \mathcal{M}^f(\Delta t)^{-1}$.

On a alors le théorème suivant, énoncé dans [71].

Théorème 1.1. *Considérons l'équation différentielle*

$$x'(t) = f(x(t)).$$

Soit $\psi(\Delta t) : E \to E$, *un opérateur symétrique en temps permettant une approximation de la solution de cette équation. Alors,* ψ *est au moins d'ordre* 2 *en temps, c'est-à-dire que pour tout* $x \in E$,

$$\psi(\Delta t)x(t) = x(t + \Delta t) + O(\Delta t^3).$$

Démonstration. *Soit*

$$x'(t) = f(x(t)).$$

Soit ψ *un opérateur symétrique en temps approchant à l'ordre* 1 *la solution de cette équation au temps* $T = Nt\Delta t$. *L'erreur commise sur chaque pas de temps* Δt *est donc de l'ordre de* Δt^2

$$\psi(\Delta t)x(t) = x(t + \Delta t) + O(\Delta t^2). \tag{2.18}$$

En effectuant un développement de Taylor en t, *nous avons*

$$\begin{aligned} x(t + \Delta t) &= x(t) + \Delta t x'(t) + \frac{\Delta t^2}{2} x''(t) + O(\Delta t^3), \\ &= x(t) + \Delta t f(x(t)) + \frac{\Delta t^2}{2} f'(x(t)) f(x(t)) + O(\Delta t^3), \end{aligned} \tag{2.19}$$

car en dérivant l'équation différentielle, nous avons $x''(t) = f'(x(t))f(x(t))$. *De plus, un développement de Taylor en* Δt *de* ψ *nous donne*

$$\psi(\Delta t)x(t) = \psi(0)x(t) + \Delta t \psi'(0)x(t) + \frac{\Delta t^2}{2} \psi''(0)x(t) + O(\Delta t^3). \tag{2.20}$$

Comme ψ *est un opérateur d'ordre* 1 (2.18), *nous pouvons identifier les termes des équations* (2.19) *et* (2.20), *nous obtenons*

$$\begin{aligned} \psi(0)x(t) &= x(t), \\ \psi'(0)x(t) &= f(x(t)). \end{aligned}$$

Pour que ψ soit un opérateur d'ordre 2, il faudrait que

$$\psi''(0)x(t) = f'(x(t))f(x(t)). \tag{2.21}$$

Comme ψ est un opérateur symétrique, nous avons

$$\psi(-\Delta t)(\psi(\Delta t)x(t)) = x(t). \tag{2.22}$$

Or, en effectuant deux développements de Taylor successifs, nous obtenons

$$\begin{aligned}
\psi(-\Delta t)(\psi(\Delta t)x(t)) &= \psi(-\Delta t)\left(x(t) + \Delta t x'(t) + \frac{\Delta t^2}{2}\psi''(0)x(t) + O(\Delta t^3)\right), \\
&= x(t) + \Delta t f(x(t)) + \frac{\Delta t^2}{2}\psi''(0)x(t) \\
&\quad - \Delta t f\left(x(t) + \Delta t f'(x(t)) + \frac{\Delta t^2}{2}\psi''(0)x(t)\right) \\
&\quad + \frac{\Delta t^2}{2}\psi''(0)\left(x(t) + \Delta t f'(x(t)) + \frac{\Delta t^2}{2}\psi''(0)x(t)\right) + O(\Delta t^3), \\
&= x(t) + \Delta t f(x(t)) + \frac{\Delta t^2}{2}\psi''(0)x(t) - \Delta t f(x(t)) \\
&\quad - \Delta t^2 f'(x(t))f(x(t)) + \frac{\Delta t^2}{2}\psi''(0)x(t) + O(\Delta t^3), \\
&= x(t) + \Delta t^2\left(\psi''(0)x(t) - f'(x(t))f(x(t))\right) + O(\Delta t^3).
\end{aligned}$$

La symétrie de l'opérateur ψ (2.22) impose donc

$$\psi''(0)x(t) = f'(x(t))f(x(t)).$$

La condition (2.21) pour obtenir un ordre de convergence de 2 est donc bien vérifiée lorsque l'opérateur ψ est symétrique.

Corollaire 1.1. *Dans le cas de la sur-relaxation $\omega = 2$, l'opérateur $\mathcal{S}^f$ est d'ordre 2 en temps.*

Définition 1.2. *Un opérateur ψ est semi-symétrique en temps si $\psi \circ \psi$ est un opérateur symétrique en temps.*

Proposition 1.2. *Lorsque $\omega = 2$, l'opérateur $\mathcal{H}^f$ est semi-symétrique en temps.*

Démonstration. *Comme $\mathcal{S}^f(\Delta t) = \mathcal{H}^f\left(\frac{\Delta t}{2}\right) \circ \mathcal{H}^f\left(\frac{\Delta t}{2}\right)$ et $\mathcal{S}^f$ est un opérateur symétrique lorsque $\omega = 2$ d'après la proposition 1.1, $\mathcal{H}^f$ est donc un opérateur semi-symétrique en temps.*

2 Vitesses cinétiques

Nous noterons $DdQn_v{}^r$ un schéma de dimension d, avec n_v vitesses cinétiques et résolvant un système de lois de conservation de dimension r. Dans le cas $r = 1$, nous allégerons cette notation en $DdQn_v$. Lorsque nous considérons un modèle à d dimensions, le vecteur cinétique

d'équilibre $\boldsymbol{f}^{eq}$ doit vérifier $d+1$ conditions de consistance (2.6). Il est donc nécessaire que le modèle cinétique possède au moins $d+1$ vitesses cinétiques, mais nous pouvons aussi considérer des modèles cinétiques avec plus de vitesses, cela permet de disposer de degrés de liberté pour obtenir un schéma avec des propriétés supplémentaires.

Dans cette section, nous présenterons les modèles cinétiques utilisés dans la suite.

2.1 Le modèle $D1Q2$

Le modèle $D1Q2$ comporte $n_v = 2$ vitesses cinétiques

$$\boldsymbol{\lambda}_1 = \begin{pmatrix} \lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_2 = \begin{pmatrix} -\lambda \end{pmatrix}.$$

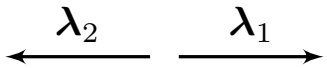

FIGURE 2.1 : Représentation des vitesses cinétiques du modèle $D1Q2$.

La matrice des moments, défini en (2.7), est alors

$$V = \begin{pmatrix} 1 & 1 \\ \lambda & -\lambda \end{pmatrix}. \tag{2.23}$$

En inversant cette matrice, nous obtenons les vecteurs cinétiques d'équilibre

$$\boldsymbol{f}_k^{eq}(\boldsymbol{w}) = \frac{\boldsymbol{w}}{2} + \frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{q}(\boldsymbol{w})}{2\lambda^2}.$$

Notons que dans ce cas, le schéma obtenu est identique au schéma de relaxation de Jin-Xin [61].

2.2 Le modèle $D2Q3$

Le modèle $D2Q3$ comporte $n_v = 3$ vitesses cinétiques, uniformément réparties sur le cercle de rayon λ

$$\boldsymbol{\lambda}_1 = \begin{pmatrix} \lambda \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_2 = \begin{pmatrix} -\frac{\lambda}{2} \\ \frac{\lambda\sqrt{3}}{2} \end{pmatrix}, \quad \boldsymbol{\lambda}_3 = \begin{pmatrix} -\frac{\lambda}{2} \\ -\frac{\lambda\sqrt{3}}{2} \end{pmatrix}.$$

La matrice des moments est donc

$$V = \begin{pmatrix} 1 & 1 & 1 \\ \lambda & -\frac{\lambda}{2} & -\frac{\lambda}{2} \\ 0 & \frac{\lambda\sqrt{3}}{2} & -\frac{\lambda\sqrt{3}}{2} \end{pmatrix}. \tag{2.24}$$

En inversant cette matrice, nous obtenons les vecteurs cinétiques d'équilibre

$$\boldsymbol{f}_k^{eq}(\boldsymbol{w}) = \frac{\boldsymbol{w}}{3} + \frac{2\boldsymbol{\lambda}_k \cdot \boldsymbol{q}(\boldsymbol{w})}{3\lambda^2}. \tag{2.25}$$

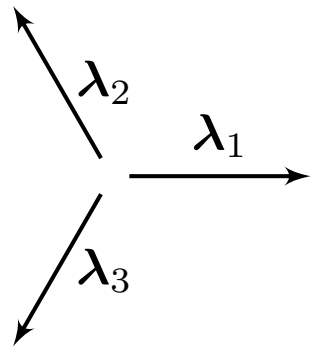

FIGURE 2.2 : Représentation des vitesses cinétiques du modèle $D2Q3$.

2.3 Le modèle $D2Q4$

Le modèle $D2Q4$ comporte $n_v = 4$ vitesses cinétiques suivant les axes cartésiens

$$\boldsymbol{\lambda}_1 = \begin{pmatrix} \lambda \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_2 = \begin{pmatrix} -\lambda \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_3 = \begin{pmatrix} 0 \\ \lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_4 = \begin{pmatrix} 0 \\ -\lambda \end{pmatrix}.$$

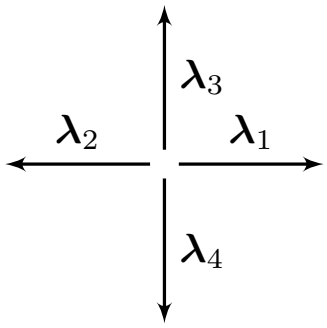

FIGURE 2.3 : Représentation des vitesses cinétiques du modèle $D2Q4$.

Comme $n_v = 4 > d + 1 = 3$, nous avons un degré de liberté supplémentaire. La dernière fonction moment est généralement définie par $m^3(\boldsymbol{\lambda}) = (\lambda^1)^2 - (\lambda^2)^2$ (voir [45]), ce qui nous donne la matrice des moments

$$V = \begin{pmatrix} 1 & 1 & 1 & 1 \\ \lambda & -\lambda & 0 & 0 \\ 0 & 0 & \lambda & -\lambda \\ \lambda^2 & \lambda^2 & -\lambda^2 & -\lambda^2 \end{pmatrix}. \tag{2.26}$$

En inversant cette matrice, nous obtenons les vecteurs cinétiques d'équilibre

$$\boldsymbol{f}_k^{eq}(\boldsymbol{w}) = \frac{\boldsymbol{w}}{4} + \frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{q}(\boldsymbol{w})}{2\lambda^2} + \frac{m^3(\boldsymbol{\lambda}_k) z_3^{eq}(\boldsymbol{w})}{4\lambda^4}. \tag{2.27}$$

Nous choisissons généralement le moment d'équilibre $\boldsymbol{z}_3^{eq} = 0$.

2.4 Le modèle $D2Q4$ twisté

Nous pouvons également considérer un modèle à $n_v = 4$ vitesses cinétiques, mais suivant les bissectrices du plan cartésien :

$$\boldsymbol{\lambda}_1 = \begin{pmatrix} \lambda \\ \lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_2 = \begin{pmatrix} -\lambda \\ \lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_3 = \begin{pmatrix} -\lambda \\ -\lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_4 = \begin{pmatrix} \lambda \\ -\lambda \end{pmatrix}.$$

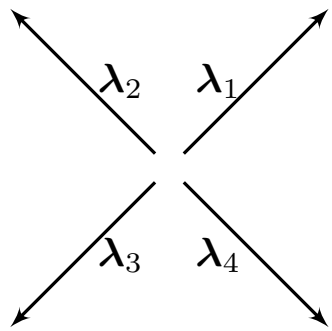

FIGURE 2.4 : Représentation des vitesses cinétiques du modèle $D2Q4$ twisté.

Nous choisissons généralement $m^3(\boldsymbol{\lambda}) = \lambda^1\lambda^2$ (voir [45]), ce qui nous donne la matrice des moments

$$V = \begin{pmatrix} 1 & 1 & 1 & 1 \\ \lambda & -\lambda & -\lambda & \lambda \\ \lambda & \lambda & -\lambda & -\lambda \\ \lambda^2 & -\lambda^2 & \lambda^2 & -\lambda^2 \end{pmatrix}.$$

En inversant cette matrice, nous obtenons les vecteurs cinétiques d'équilibre

$$\boldsymbol{f}_k^{eq} = \frac{\boldsymbol{w}}{4} + \frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{q}(\boldsymbol{w})}{4\lambda^2} + \frac{m^3(\boldsymbol{\lambda}_k) z_3^{eq}(\boldsymbol{w})}{4\lambda^4}.$$

2.5 Le modèle $D2Q5$

Le modèle $D2Q5$ comporte les mêmes vitesses cinétiques que le modèle $D2Q4$ classique, auxquelles nous ajoutons une cinquième vitesse nulle.

$$\boldsymbol{\lambda}_1 = \begin{pmatrix} \lambda \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_2 = \begin{pmatrix} -\lambda \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_3 = \begin{pmatrix} 0 \\ \lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_4 = \begin{pmatrix} 0 \\ -\lambda \end{pmatrix}, \quad \boldsymbol{\lambda}_5 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Nous utiliserons le modèle $D2Q5$ uniquement dans le chapitre 7. Nous considérerons alors un flux linéaire $\boldsymbol{q}(\boldsymbol{w}) = \boldsymbol{v}\boldsymbol{w}$. Nous définirons les équilibres cinétiques par

$$\boldsymbol{f}_k^{eq} = \boldsymbol{w}\left(\boldsymbol{\lambda}_k \cdot \boldsymbol{v}\right)_+ \qquad \text{pour } k = 1, 2, 3, 4,$$

et

$$\boldsymbol{f}_5^{eq} = \boldsymbol{w} - \sum_{k=1}^{4} \boldsymbol{f}_k^{eq},$$

avec $(x)_+ = \max\{x, 0\}$ la partie positive.

3 Conclusion

Dans ce chapitre, nous avons rappelé les principes de l'approximation d'un système de lois de conservation par un modèle cinétique vectoriel.

Afin de calculer la solution du système cinétique, nous avons vu qu'il était préférable d'utiliser un schéma de *splitting* qui permet de découpler le transport et la relaxation. Cette approche permet de plus de construire très facilement un schéma d'intégration d'ordre 2 en temps grâce à la sur-relaxation.

Enfin, nous avons passé en revue les modèles cinétiques les plus courants : $D1Q2$, $D2Q4$, *etc.* Dans le chapitre suivant, nous allons étudier plus précisément le lien entre le modèle cinétique et le système de lois de conservation. Pour cela nous utiliserons la technique de l'équation équivalente. Cette technique permet d'établir en un certain sens la consistance de l'approximation cinétique.

Chapter 3

Equivalent system in the general case

We have constructed in the previous chapter a general kinetic method for approximating system of the conservation laws

$$\partial_t \boldsymbol{w} + \nabla \cdot \boldsymbol{q}(\boldsymbol{w}) = 0. \tag{3.1}$$

We wish to analyze the solution of this system given by the kinetic relaxation scheme $DdQn_v$, described in chapter 2.

To do that, we want to compute the equivalent equation of the solution of (3.1) approximated by the symmetric operator $\mathcal{S}^f$ defined by 2.17 (p.14), not only on the variable $\boldsymbol{w}$, but on the full set of n_v kinetic variables $(\boldsymbol{w}, \boldsymbol{Y})$, with $\boldsymbol{Y}$ the flux error that we will define later.

The method of equivalent equation is a general method for proving the consistency of numerical schemes with the approximated system. It is generally based on simple but tedious Taylor expansions. It not only allows to prove the consistency of the scheme, but also explains in some cases the behavior of the numerical scheme by inspecting the higher order terms of the expansion. Generally, the stability of the numerical scheme relies on an equivalent equation containing second order diffusive terms that are not present in the original system. The behavior of classical schemes such as the Rusanov or McCormak schemes [70, 80] is often explained in this way. In a very cited review, Harten Lax and van Leer linked the numerical viscosity with entropy properties of the numerical scheme and introduced the famous HLL scheme [54]. In the framework of first order hyperbolic conservations the equivalent equation technique is exposed for instance in the book of Leveque [65]. While the equivalent equation gives important information on the consistency of the scheme, one must be very careful for analyzing the stability as it is explained in [30]. Because of its importance, the equivalent equation technique has been applied to the study of the kinetic approximation. We refer for instance to the works of Aregba-Natalini [6], Dubois [34], Dellar [28], Graille [50], Drui & al. [33], Courtès & al. [24].

A difficulty of the analysis of kinetic models is that they mix two features: additional hidden variables (the scheme contains $(\boldsymbol{w}, \boldsymbol{Y})$ instead of $\boldsymbol{w}$ only), and also a stiff relaxation behavior. In [34, 24, 50], the analysis of the stiff relaxation and the Taylor expansion are performed at the same time. This gives the good result, but does not allow to understand the separate

effects easily. Therefore in this work we prefer to first perform the Taylor expansion, and then the Chapman-Enskog analysis of the stiff terms. A special case is the case of an over-relaxation with $\omega = 2$. In this case, the Taylor expansion is sufficient because the stiff terms vanish. The case $\omega \neq 2$ will be treated in the next chapter.

In this chapter, we are thus first interested in the equivalent equation for the particular case of the relaxation parameter $\omega = 2$. Indeed, in this case, the stiff term vanishes and the operator $\mathcal{S}^f$ appears to be symmetric in time, and therefore, gives an approximation of the solution of order 2 in time. We will analyze both one-dimensional and two-dimensional kinetic models. We will detail the computation of the equivalent equation in these cases, and will apply it for different kinetic models. Then, we will perform numerical experiments in order to quantify the adequacy of the consistency analysis with real practical cases.

1 Computation of the equivalent system

In order to analyze the effect of the kinetic over-relaxation on the initial variable $\boldsymbol{w}$ of the hyperbolic system (3.1), we introduce additional quantities $\boldsymbol{Z}$, $\boldsymbol{z}_k$ for $k = 0, \ldots, n_v - 1$, such that $\boldsymbol{z}_0 = \boldsymbol{w}$, called the *approximated moments*, defined by

$$\boldsymbol{Z} = \begin{pmatrix} \boldsymbol{z}_0 \\ \vdots \\ \boldsymbol{z}_{n_v-1} \end{pmatrix} = V \begin{pmatrix} \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_{n_v} \end{pmatrix}. \tag{3.2}$$

We also introduce the *flux error* $\boldsymbol{y}^i$ defined by

$$\boldsymbol{y}_i = \boldsymbol{z}_k - \boldsymbol{z}_k^{eq}, \quad k = 0, \cdots, n_v - 1, \quad \boldsymbol{Y} = \begin{pmatrix} \boldsymbol{y}_0 \\ \vdots \\ \boldsymbol{y}_{n_v-1} \end{pmatrix}. \tag{3.3}$$

Because of (3.2) and (2.8) we find that

$$\boldsymbol{Y} = V(\boldsymbol{F} - \boldsymbol{F}^{eq}(\boldsymbol{w})).$$

Let us insist that, with our definition, while $\boldsymbol{Y}$ has r rows and n_v columns, it lives in a $(r \times (n_v - 1))$-dimensional space because we always have

$$\boldsymbol{y}_0 = 0.$$

The main point to compute the equivalent system is then to rewrite the LBM algorithm in the $(\boldsymbol{w}, \boldsymbol{Y})$ variables. With the above notations, we can rewrite the transport-relaxation algorithm as follows. We start, with a macroscopic field $\boldsymbol{w}(\boldsymbol{x}, 0)$ and a flux error field $\boldsymbol{Y}(\boldsymbol{x}, 0)$ at time $t = 0$. From this flux error, we can compute

$$\boldsymbol{Z} = \boldsymbol{Y} + \boldsymbol{Q}(\boldsymbol{w}),$$

and then

$$\boldsymbol{F} = V^{-1}\boldsymbol{Z}.$$

Then we can transport the components of $\boldsymbol{F}$ during a time step Δt

$$\boldsymbol{f}_k(\boldsymbol{x}, \Delta t) = \boldsymbol{f}_k(\boldsymbol{x} - \Delta t \boldsymbol{\lambda}_k, 0). \tag{3.4}$$

This defines the transport operator $\mathcal{T}^f(\Delta t)$ acting on the initial field $\boldsymbol{x} \mapsto \boldsymbol{F}(\boldsymbol{x}, 0)$. More precisely, $\mathcal{T}^f$ is unambiguously defined by (3.4), (2.9) and

$$(\mathcal{T}^f(\Delta t)\boldsymbol{F}(\cdot, 0))(\boldsymbol{x}) = \boldsymbol{F}(\boldsymbol{x}, \Delta t).$$

Finally, we return back to the $(\boldsymbol{w}, \boldsymbol{Y})$ variables by

$$\boldsymbol{w} = \boldsymbol{F} \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}, \quad \boldsymbol{Y} = V\boldsymbol{F} - \boldsymbol{Q}(\boldsymbol{w}).$$

In the $(\boldsymbol{w}, \boldsymbol{Y})$ variables, the transport step can thus be expressed in the following operator form

$$\begin{aligned} \mathcal{T}(\Delta t) \begin{pmatrix} \boldsymbol{w}(\cdot, 0) \\ \boldsymbol{Y}(\cdot, 0) \end{pmatrix} = & \begin{pmatrix} \left\{\mathcal{T}^f(\Delta t)\left[V^{-1}\left(\boldsymbol{Y}(\cdot, 0) + \boldsymbol{Q}(\boldsymbol{w}(\cdot, 0))\right)\right]\right\} \cdot \mathbb{1} \\ V\left\{\mathcal{T}^f(\Delta t)\left[V^{-1}\left(\boldsymbol{Y}(\cdot, 0) + \boldsymbol{Q}(\boldsymbol{w}(\cdot, 0))\right)\right]\right\} \end{pmatrix} \\ & - \begin{pmatrix} 0 \\ \boldsymbol{Q}\left(\left\{\mathcal{T}^f(\Delta t)\left[V^{-1}\left(\boldsymbol{Y}(\cdot, 0) + \boldsymbol{Q}(\boldsymbol{w}(\cdot, 0))\right)\right]\right\} \cdot \mathbb{1}\right) \end{pmatrix}. \end{aligned} \tag{3.5}$$

In the $(\boldsymbol{w}, \boldsymbol{Y})$ set of variables, the relaxation step (2.13) is very simple. It writes

$$\mathcal{R}_\omega \begin{pmatrix} \boldsymbol{w}(\cdot, t)) \\ \boldsymbol{Y}(\cdot, t)) \end{pmatrix} = \begin{pmatrix} \boldsymbol{w}(\cdot, t) \\ (1-\omega)\boldsymbol{Y}(\cdot, t) \end{pmatrix}, \tag{3.6}$$

thus $\boldsymbol{w}$ is continuous in time.

In the case of the over-relaxation $\omega = 2$, it becomes

$$\mathcal{R}_\omega \begin{pmatrix} \boldsymbol{w}(\cdot, t)) \\ \boldsymbol{Y}(\cdot, t)) \end{pmatrix} = \begin{pmatrix} \boldsymbol{w}(\cdot, t) \\ -\boldsymbol{Y}(\cdot, t) \end{pmatrix}, \tag{3.7}$$

thus $\boldsymbol{Y}$ oscillates around zero at the frequency $1/\Delta t$.

Formula (3.5) and (3.6) define the LBM operator $\mathcal{M}$ in the $(\boldsymbol{w}, \boldsymbol{Y})$ variables such as

$$\begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix}(\cdot, \Delta t) = \mathcal{M}(\Delta t) \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix}(\cdot, 0),$$

with

$$\mathcal{M}(\Delta t) = \mathcal{R}_\omega \circ \mathcal{T}(\Delta t). \tag{3.8}$$

We can also redefine the LBM half-symmetric operator $\mathcal{H}$ with

$$\mathcal{H}(\Delta t) = \mathcal{T}\left(\frac{\Delta t}{2}\right) \circ \mathcal{R}_\omega \circ \mathcal{T}\left(\frac{\Delta t}{2}\right).$$

In the analysis, it is better to cancel the oscillations (3.7) in the flux error $\boldsymbol{Y}$. Instead of analyzing the LBM operator directly, we will use the symmetric operator $\mathcal{S}$ defined in (2.17) (p.14) as

$$\mathcal{S}(\Delta t) = \mathcal{H}(\Delta t/2) \circ \mathcal{H}(\Delta t/2).$$

This amounts to only observing the LBM discrete solution at the even time steps. Then estimating the time derivative of $(\boldsymbol{w}, \boldsymbol{Y})$ by a centered finite difference, we have, on the one hand

$$\partial_t \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix} (\cdot, t) = \frac{\begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix} (\cdot, t + \Delta t) - \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix} (\cdot, t - \Delta t)}{2\Delta t} + O(\Delta t^2),$$

that we can write with the operator $\mathcal{S}$

$$\partial_t \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix} (\cdot, t) = \frac{\mathcal{S}(\Delta t) - \mathcal{S}^{-1}(\Delta t)}{2\Delta t} \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix} (\cdot, t) + O(\Delta t^2).$$

On the other hand, we can perform a Taylor expansion of

$$\frac{\mathcal{S}(\Delta t) - \mathcal{S}^{-1}(\Delta t)}{2\Delta t} \begin{pmatrix} \boldsymbol{w} \\ \boldsymbol{Y} \end{pmatrix} (\cdot, t),$$

with respect to Δt.

The proof of consistency is based on standard Taylor expansions. The approach is classical in the analysis of the Lattice Boltzmann Method (LBM). See also for instance [34, 75, 33]. The calculations can be performed with Maple, using the code given in Annex B (p.169). The equivalent system will be given in Section 2, for different kinetic models.

We will also see that the above analysis allows recovering formally the so-called sub-characteristic condition. We will prove that, in general, the equivalent system is hyperbolic under the condition that the norms $|\boldsymbol{\lambda}_k|$ are large enough.

2 Applications

From now on, we consider a one-dimensional unknown function $w \in \mathbb{R}$, namely $r = 1$. Thus, we are interested by the single conservation law

$$\partial_t w + \sum_{i=1}^{d} \partial_i q_i(w) = 0.$$

This is sufficient for the application that we have in mind: the transport models in plasma physics. We are going to apply the computation of the equivalent equation described in Section 1 to different kinetic models. We remind that in this chapter, we only consider the case of the relaxation parameter $\omega = 2$.

2.1 Equivalent system of the $D1Q2$ model

The equivalent system of the $D1Q2$ model has already been computed by Drui & *al.* in [33]. In this case, the equivalent system of equations is

$$\partial_t \begin{pmatrix} w \\ y \end{pmatrix} + \begin{pmatrix} q'(w) & 0 \\ 0 & -q'(w) \end{pmatrix} \partial_x \begin{pmatrix} w \\ y \end{pmatrix} = O(\Delta t^2). \tag{3.9}$$

As stated in [33], the equivalent system is hyperbolic independently of the chosen value for λ. For recovering the sub-characteristic condition, it is necessary to analyze the higher order terms in Δt.

2.2 Equivalent system of the $D2Q3$ model

Theorem 2.1. *The equivalent system of the D2Q3 model described in Subsection 2.2 (p.17) is*

$$\begin{aligned} \partial_t \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} &+ \begin{pmatrix} q_1'(w) & 0 & 0 \\ 0 & \frac{\lambda}{2} - q_1'(w) & 0 \\ 0 & -q_2'(w) & \frac{-\lambda}{2} \end{pmatrix} \partial_1 \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} \\ &+ \begin{pmatrix} q_2'(w) & 0 & 0 \\ 0 & 0 & -\frac{\lambda}{2} - q_1'(w) \\ 0 & \frac{-\lambda}{2} & -q_2'(w) \end{pmatrix} \partial_2 \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} = O(\Delta t^2). \end{aligned} \tag{3.10}$$

Proof. We apply the computation described in Subsection 1 to the $D2Q3$ model.

Using the Maple code in Annex B (p.169), we obtain the Taylor expansion of the symmetric operator $\mathcal{S}$

$$\left(\frac{\mathcal{S}(\Delta t) - \mathcal{S}^{-1}(\Delta t)}{2\Delta t} \right) \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} -q_1'(w)\partial_1 w - q_2'(w)\partial_2 w + O(\Delta t^2) \\ (q_1'(w) - \frac{\lambda}{2})\partial_1 y_1 + (q_1'(w) + \frac{\lambda}{2})\partial_2 y_2 + O(\Delta t^2) \\ q_2'(w)\partial_1 y_1 + \frac{\lambda}{2}\partial_1 y_2 + \frac{\lambda}{2}\partial_2 y_1 + q_2'(w)\partial_2 y_2 + O(\Delta t^2) \end{pmatrix}.$$

This gives us the expected equivalent system. □

Remark 2.1. *We can remark that, up to $O(\Delta t^2)$ terms, the first line of the system corresponds to our initial equation (3.1). We can also observe that, up to $O(\Delta t^2)$ terms, the dynamics of w is independent of the flux errors Y. It means we do not need to choose Y small to obtain a good approximation of w.*

2.3 Equivalent system of the $D2Q4$ model

In the $D2Q4$ model, we solve $n_v = 4$ kinetic equations (2.3) with $n_v = 4$ kinetic unknown $\boldsymbol{f}_k$. Therefore, we want an equivalent system with 4 equations and 4 unknowns. As for the previous model, we consider the equivalent system on w and the flux errors y_i, but we need to add a fourth unknown z_3.

Theorem 2.2. *The equivalent system of the $D2Q4$ model described in Subsection 2.3 (p.18) is*

$$\begin{aligned}\partial_t \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} &+ \begin{pmatrix} q_1'(w) & 0 & 0 & 0 \\ 0 & -q_1'(w) & 0 & \frac{1}{2} \\ 0 & -q_2'(w) & 0 & 0 \\ 0 & \lambda^2 & 0 & 0 \end{pmatrix} \partial_1 \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} \\ &+ \begin{pmatrix} q_2'(w) & 0 & 0 & 0 \\ 0 & 0 & -q_1'(w) & 0 \\ 0 & 0 & -q_2'(w) & -\frac{1}{2} \\ 0 & 0 & -\lambda^2 & 0 \end{pmatrix} \partial_2 \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} = O(\Delta t^2).\end{aligned} \tag{3.11}$$

Proof. We apply the computation described in Subsection 1 to the $D2Q4$ model. Using the Maple code in Annex B (p.169), we obtain the Taylor expansion of the symmetric operator $\mathcal{S}$

$$\left(\frac{\mathcal{S}(\Delta t) - \mathcal{S}^{-1}(\Delta t)}{2\Delta t}\right) \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} -q_1'(w)\partial_1 w - q_2'(w)\partial_2 w + O(\Delta t^2) \\ q_1'(w)\partial_1 y_1 + q_1'(w)\partial_2 y_2 - \frac{1}{2}\partial_1 z_3 + O(\Delta t^2) \\ q_2'(w)\partial_1 y_1 + q_2'(w)\partial_2 y_2 + \frac{1}{2}\partial_2 z_3 + O(\Delta t^2) \\ -\lambda^2 \partial_1 y_1 + \lambda^2 \partial_2 y_2 + O(\Delta t^2) \end{pmatrix}.$$

This gives us the expected equivalent system. □

2.4 Equivalent system of the $D2Q4$ twisted model

Theorem 2.3. *The equivalent system of the $D2Q4$ twisted model described in Subsection 2.4 (p.19) is*

$$\begin{aligned}\partial_t \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} &+ \begin{pmatrix} q_1'(w) & 0 & 0 & 0 \\ 0 & -q_1'(w) & 0 & 0 \\ 0 & -q_2'(w) & 0 & 1 \\ 0 & 0 & \lambda^2 & 0 \end{pmatrix} \partial_1 \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} \\ &+ \begin{pmatrix} q_2'(w) & 0 & 0 & 0 \\ 0 & 0 & -q_1'(w) & 1 \\ 0 & 0 & -q_2'(w) & 0 \\ 0 & \lambda^2 & 0 & 0 \end{pmatrix} \partial_2 \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} = O(\Delta t^2).\end{aligned} \tag{3.12}$$

Proof. We apply the computation described in Subsection 1 to the $D2Q4$ twisted model. Using the Maple code in Annex B (p.169), we obtain the Taylor expansion of the symmetric operator $\mathcal{S}$

$$\left(\frac{\mathcal{S}(\Delta t) - \mathcal{S}^{-1}(\Delta t)}{2\Delta t}\right) \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} -q_1'(w)\partial_1 w - q_2'(w)\partial_2 w + O(\Delta t^2) \\ q_1'(w)\partial_1 y_1 + q_1'(w)\partial_2 y_2 - \partial_2 z_3 + O(\Delta t^2) \\ q_2'(w)\partial_1 y_1 + q_2'(w)\partial_2 y_2 - \partial_1 z_3 + O(\Delta t^2) \\ -\lambda^2 \partial_1 y_2 - \lambda^2 \partial_2 y_1 + O(\Delta t^2) \end{pmatrix}.$$

This gives us the expected equivalent system. □

3 Numerical validation of the equivalent systems

Now, we want to validate numerically the equivalent systems that we have obtained in Section 1, in the case of a linear flux. We want to confirm that the solution of the equivalent system is quantitatively close to the numerical solution. For this purpose, we are going to compare the flux errors, which are the solution of our equivalent system for a given kinetic scheme, with those computed from the solution of the initial equation (3.1) obtained with the same kinetic model.

In practice, as Y is the error between the flux $\boldsymbol{q}(w)$ and the approximated flux $\boldsymbol{z}$, we expect Y to be close to 0. However, this is not a mandatory condition. Here, we are going to choose Y with the same order of magnitude as w.

3.1 Validation of the equivalent system of the $D2Q3$ model

First, we solve the system (3.1) using the $D2Q3$ kinetic scheme. Then, we can compute the flux error $Y^{kin} = Z^{kin} - \boldsymbol{q}(w)$, with $Z^{kin} = \sum_{k=1}^{n_v} \boldsymbol{\lambda}_k f_k$.

Secondly, we solve the equivalent system

$$\partial_t Y + \begin{pmatrix} \frac{\lambda}{2} - q_1'(w) & 0 \\ -q_2'(w) & -\frac{\lambda}{2} \end{pmatrix} \partial_1 Y + \begin{pmatrix} 0 & -\frac{\lambda}{2} - q_1'(w) \\ -\frac{\lambda}{2} & -q_2'(w) \end{pmatrix} \partial_2 Y = 0, \tag{3.13}$$

with a high order finite volume method, for instance. We note Y^{vf} the solution.

We can then compare the Y^{kin} obtained from the computation of the hyperbolic equation and Y^{vf} obtained by solving the equivalent systems (3.13): as the equivalent system is an approximation of the system solved by the $D2Q3$ model, Y^{vf} is supposed to converge toward Y^{kin} when Δx and Δt converge to 0.

We consider a linear flux $\boldsymbol{q}(w) = \begin{pmatrix} v_1 w \\ v_2 w \end{pmatrix}$, where the velocities v_1 and v_2 are constant: $v_1 = 1$ and $v_2 = 1$. We choose the norm of kinetic velocities: $\lambda = 3$. Let us remark that this choice satisfy the sub-characteristic condition that we will define later, in Subsection 2.3.

We consider a square geometry $\Omega = [0,1] \times [0,1]$. We initialize the density and the flux error with the Gaussian functions

$$w(\boldsymbol{x}, 0) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}_0^w\|^2}{2\sigma^2}\right) \qquad \text{and} \qquad y_i(\boldsymbol{x}, 0) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}_0^y\|^2}{2\sigma^2}\right),$$

with $\sigma = 0.05$, $\boldsymbol{x}_0^w = (0.25, 0.25)$ and $\boldsymbol{x}_0^y = (0.5, 0.5)$. We compare the solutions at time $T = 0.06$.

We can observe in Table 3.1, that the L_2 errors between Y^{kin} and Y^{vf} are small and decrease when we refine the mesh. This confirms that the solution Y^{vf} of the equivalent equation for the $D2Q3$ model converges towards the flux error of the initial equation Y^{kin}.

3.2 Validation of the equivalent system of the $D2Q4$ model

Now, we want to validate numerically the equivalent system of the $D2Q4$ model. As for the validation of the $D2Q3$ equivalent equation in Subsection 3.1, we compare the flux error

Mesh size	$\|y_1^{vf} - y_1^{kin}\|$	$\|y_2^{vf} - y_2^{kin}\|$
100×100	2.57164×10^{-3}	1.92973×10^{-2}
200×200	1.90878×10^{-3}	1.06497×10^{-2}
400×400	6.60124×10^{-4}	4.23034×10^{-3}
800×800	5.64567×10^{-4}	1.95625×10^{-3}

Table 3.1: L_2 errors between the flux error computed from the hyperbolic equation solved with the $D2Q3$ kinetic scheme and the flux error from the equivalent system, for different mesh refinements.

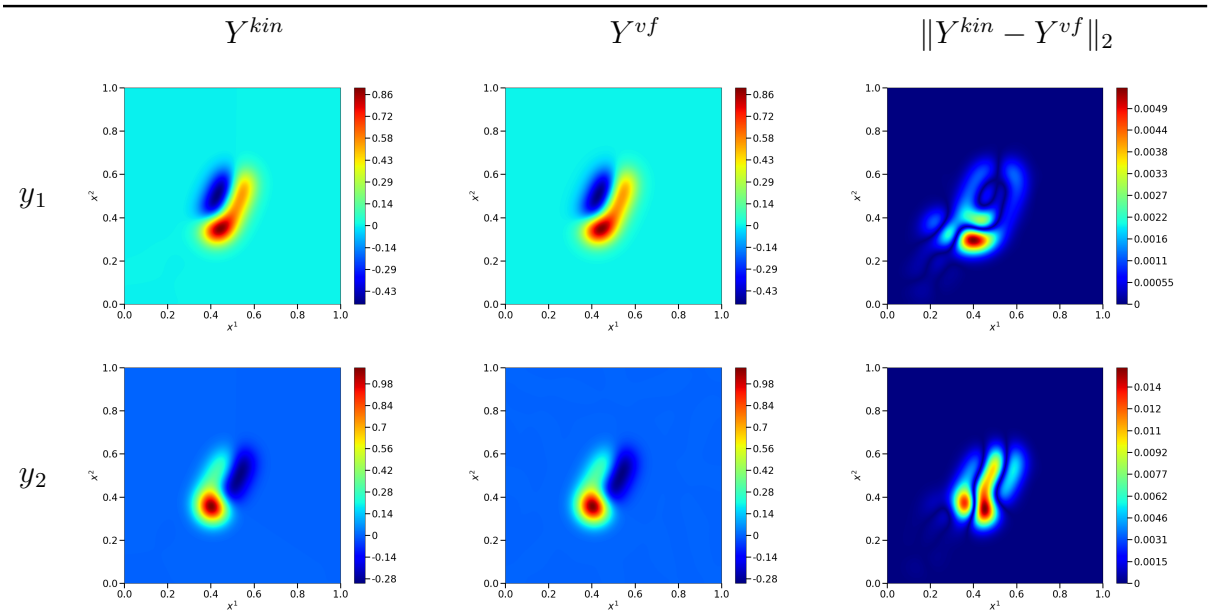

Table 3.2: Flux errors Y^{kin} and Y^{vf} and the norm of their difference $\|Y^{kin} - Y^{vf}\|$ at $T = 0.06$ for a mesh of size 800×800.

Y^{kin} obtained by solving the hyperbolic system (3.1) with the $D2Q4$ kinetic scheme with the solution Y^{vf} of the equivalent equation.

By using the same parameters as for the validation of the $D2Q3$ equivalent system, we obtain the flux errors Y^{kin} and Y^{vf} in Table 3.4. We can observe that Y^{kin} and Y^{vf} are visually similar and the error $\|Y^{kin} - Y^{vf}\|$ is small.

Mesh size	$\|y_1^{vf} - y_1^{kin}\|$	$\|y_2^{vf} - y_2^{kin}\|$	$\|z_3^{vf} - z_3^{kin}\|$
100×100	7.78988×10^{-4}	1.00352×10^{-3}	2.86871×10^{-3}
200×200	1.95075×10^{-4}	2.51593×10^{-4}	7.18457×10^{-4}
400×400	4.87890×10^{-5}	6.29419×10^{-5}	1.79698×10^{-4}
800×800	1.21999×10^{-5}	1.57384×10^{-5}	4.49337×10^{-5}

Table 3.3: L_2 errors between the flux error and z_3 computed with the kinetic model $D2Q4$ and the one computed from the equivalent systems, for different mesh sizes.

We can observe in Table 3.3, that the L_2 errors between Y^{kin} and Y^{vf} are small and decrease when we refine the mesh. This confirms that the solution Y^{vf} of the equivalent system

converges towards the solution Y^{kin} of the hyperbolic equation (3.1).

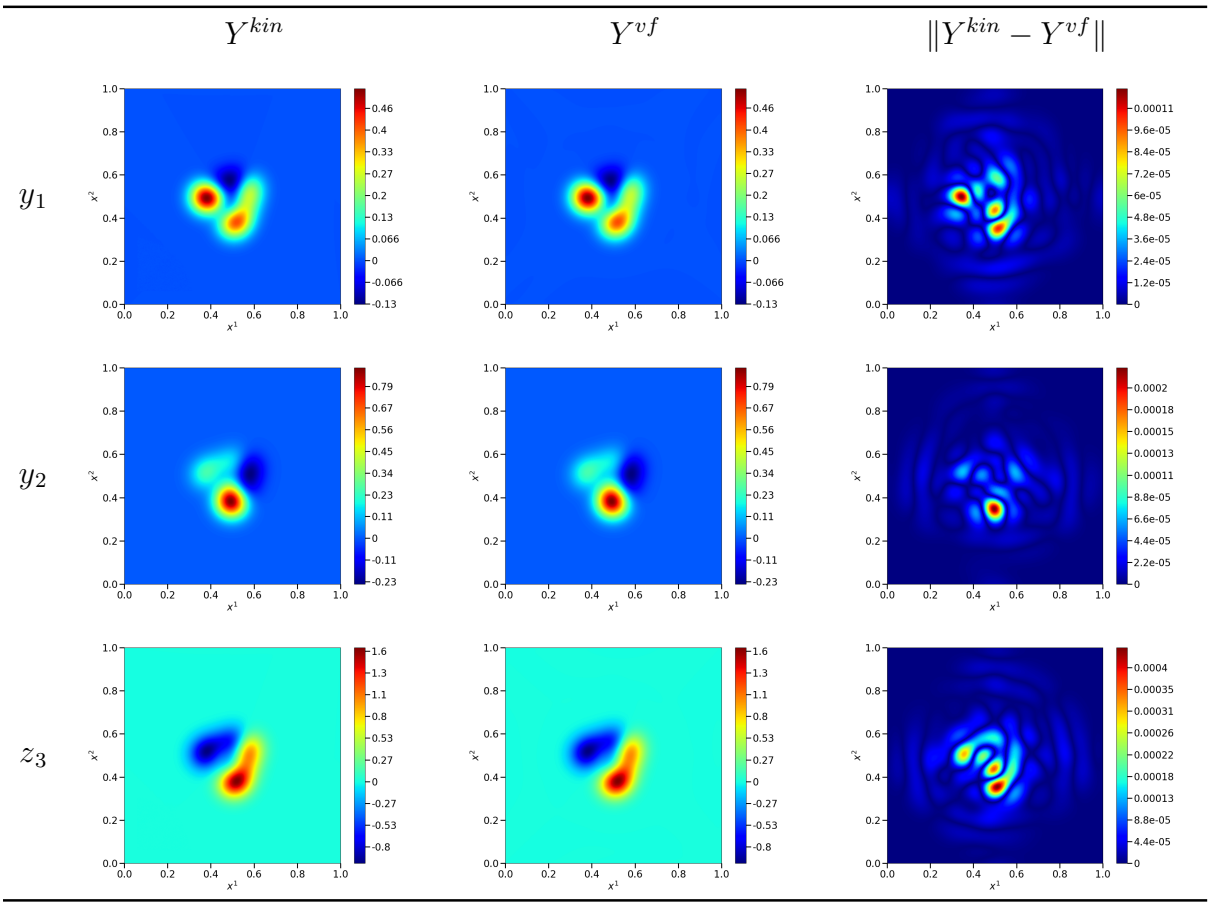

Table 3.4: Flux errors Y^{kin} and Y^{vf} and the norm of their difference $\|Y^{kin} - Y^{vf}\|$ at $T = 0.06$ for a mesh of size 800×800.

4 Conclusion

In this chapter, we have detailed our methodology to compute the equivalent system of any kinetic split scheme when $\omega = 2$. The important facts are that:

- the computations are fully automatic and are actually performed thanks to a Computer Algebra System (a CAS, Maple in our case);
- in the case $\omega = 2$, if one groups two successive time steps, the discrete operator is time symmetric. This implies that the stiff terms vanish in the analysis, which becomes easier.
- Still in the case $\omega = 2$, it is not necessary to assume the smallness of the flux error to achieve consistency. The flux error Y oscillates around zero without damping, but the conservative variables w are anyway approximated at order 2. It is confirmed by our numerical experiments.

We have computed the equivalent system for different kinetic models. Then, we have numerically validated these equations. We have focalized our study in the case of the relaxation

parameter $\omega = 2$.

In the following, in Chapter 4, we will consider any possible relaxation $1 \leqslant \omega \leqslant 2$. The consistency analysis is a little bit more difficult, because the stiff relaxation term does not vanish anymore.

Chapter 4

Application to the transport equation

Previously, in Chapter 3, we have computed the equivalent system of the split kinetic system in the case of an over-relaxation with $\omega = 2$. We have seen that it is convenient to rewrite the kinetic system in the (w, Y) variables. We have then shown that the kinetic system provides a second-order approximation of the system of conservation of laws (3.1)

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = 0,$$

whatever the size of the flux error Y.

In this chapter, we want to generalize the equivalent equation analysis for all relaxation parameters $1 \leqslant \omega < 2$. In order to simplify the computation, we shall consider a linear flux $\boldsymbol{q}(w) = \boldsymbol{v}w$. Thus, we consider the transport equation

$$\partial_t w + \sum_{i=1}^{d} v_i \partial_i w = 0.$$

But we point out that the method can be extended to any non-linear system.

Actually, we shall see that we can provide two different equivalent equations. In a first step, with the same Taylor expansion approach as in the previous chapter we get an equivalent system involving both w and Y with stiff relaxation terms in $1/\Delta t$. In a second step, from this equivalent system, we can apply a Chapman-Enskog expansion to get a new equivalent equation involving only w.

The advantage of this approach is that it is quite simple and fully "algorithmic". It also provides equivalent equations with only first-order derivatives in time. We are aware of other works in the literature that provide algorithms to obtain equivalent equations with higher order time derivatives, see for instance [49, 10].

1 Equivalent system on (w, Y)

We approximate the time derivative of (w, Y) by the centered finite difference scheme

$$\partial_t \begin{pmatrix} w \\ Y \end{pmatrix} (\cdot, t) = \frac{\begin{pmatrix} w \\ Y \end{pmatrix} (\cdot, t + \Delta t) - \begin{pmatrix} w \\ Y \end{pmatrix} (\cdot, t - \Delta t)}{2\Delta t} + O(\Delta t^2). \tag{4.1}$$

Let us denote the time evolution operator ψ, which can be $\mathcal{M}$, $\mathcal{H}$ or $\mathcal{S}$ defined in (2.15)-(2.16)-(2.17). This means that we first take $w(\cdot, t + \Delta t) = \psi(\Delta t) w(\cdot, t)$, then, we have

$$\partial_t \begin{pmatrix} w \\ Y \end{pmatrix} (\cdot, t) = \frac{\psi(\Delta t) - \psi^{-1}(\Delta t)}{2\Delta t} \begin{pmatrix} w \\ Y \end{pmatrix} (\cdot, t) + O(\Delta t^2). \tag{4.2}$$

Then, we can perform a Taylor expansion of the right-hand side of this expression (using, for instance, the Maple code given in the Annex B (p.169)). We obtain an equivalent system of the form

$$\partial_t \begin{pmatrix} w \\ Y \end{pmatrix} - \frac{a}{\Delta t} \begin{pmatrix} 0 \\ Y \end{pmatrix} + \sum_{i=1}^{d} B_i \partial_i \begin{pmatrix} w \\ Y \end{pmatrix} + \Delta t \sum_{i=1}^{d} \sum_{j=1}^{d} C_{i,j} \partial_{i,j} \begin{pmatrix} w \\ Y \end{pmatrix} = O(\Delta t^2), \tag{4.3}$$

where a and the coefficients of the matrices B_i and $C_{i,j}$ are depending on the chosen time-evolution operator and kinetic velocities.

We obtain equivalent systems with the same form as those obtained in Chapter 3 for the over-relaxation case, but with an additional damping term $-\frac{a}{\Delta t}$ on the equation on Y. It corresponds to a fast relaxation of Y toward 0 when $\omega < 2$, while the amplitude of Y did not decrease in the case of $\omega = 2$.

In the next section, we show how to perform a Chapman-Enskog expansion of this system to get a new equivalent equation involving only w.

2 Equivalent equation on w

In the case $\omega < 2$, because of the damping, we can assume that Y will be small, for instance $Y = O(\Delta t)$. It means that it exists $\tilde{Y}$ such as $Y = \tilde{Y} \Delta t$, with $\tilde{Y} = O(1)$. Inserting this expression of Y in equation 4.3 and keeping the constant terms, we obtain the expression for Y

$$y_k = \frac{\Delta t}{a} \sum_{i=1}^{d} B_i[k, 1] \partial_i w + O(\Delta t^2), \qquad \text{for } k = 1, \ldots, n_v - 1. \tag{4.4}$$

The first equation of the equivalent system (4.3) is

$$\partial_t w + \sum_{i=1}^{d} v_i \partial_i w + \sum_{i=1}^{d} \sum_{k=1}^{n_v - 1} B_i[1, k] \partial_i y_k + \Delta t \sum_{i=1}^{d} \sum_{j=1}^{d} C_{i,j}[1, 1] \partial_{i,j} w = O(\Delta t^2),$$

and thus replacing y_k by the expression (4.4), we obtain

$$\partial_t w + \sum_{i=1}^{d} v_i \partial_i w + \Delta t \sum_{i=1}^{d} \sum_{j=1}^{d} \left(\frac{1}{a} \sum_{k=1}^{n_v - 1} B_i[1, k] B_j[k, 1] + C_{i,j}[1, 1] \right) \partial_{i,j} w = O(\Delta t^2).$$

We obtain an equivalent equation that only depends on w.

3 Equivalent system and equation of the $D1Q2$ model with the operator $\mathcal{M}$

In this section, we apply the general approach presented above to particular kinetic models presented in Chapter 2.

Let us first consider the $D1Q2$ model and the operator $\mathcal{M}$, defined in (3.8) (p.23), which corresponds to just performing a transport and a relaxation steps during one time step

$$\begin{pmatrix} w \\ Y \end{pmatrix}(\cdot, t+\Delta t) = \mathcal{M}(\Delta t)\begin{pmatrix} w \\ Y \end{pmatrix}(\cdot, t),$$

with $\mathcal{M}(\Delta t) = \mathcal{R}_\omega \circ \mathcal{T}(\Delta t)$. By computing the Taylor expansion of (4.2), using the Maple code given in Annex B (p.169), we obtain the equivalent system on (w, y)

$$\begin{aligned} \partial_t \begin{pmatrix} w \\ y \end{pmatrix} - \frac{1}{\Delta t}\begin{pmatrix} 0 \\ \frac{\omega(2-\omega)}{2(\omega-1)} y \end{pmatrix} + \begin{pmatrix} v & \frac{\omega-2}{2(\omega-1)} \\ \frac{(\lambda^2-v^2)(2-\omega)}{2} & \frac{v(\omega^2-2\omega+2)}{2(\omega-1)} \end{pmatrix} \partial_x \begin{pmatrix} w \\ y \end{pmatrix} \\ + \Delta t \begin{pmatrix} 0 & 0 \\ 0 & \frac{\lambda^2\omega(\omega-2)}{4(\omega-1)} \end{pmatrix} \partial_{xx} \begin{pmatrix} w \\ y \end{pmatrix} = O(\Delta t^2). \end{aligned} \tag{4.5}$$

Now, let us assume that $y = O(\Delta t)$, and that it exists $\tilde{y} = O(1)$ such as $y = \Delta t \tilde{y}$. By replacing y by this expression in (4.5) and keeping the constant term, we obtain

$$\frac{\omega(2-\omega)}{2(\omega-1)}\tilde{y} = \frac{(\lambda^2-v^2)(2-\omega)}{2}\partial_x w + O(\Delta t).$$

It gives us

$$y = \frac{(\lambda^2-v^2)(\omega-1)}{\omega}\Delta t \partial_x w + O(\Delta t^2).$$

By reinjecting this expression of y in the first equation of the equivalent system (4.5), we obtain

$$\partial_t w + v\partial_x w + \frac{\omega-2}{2(\omega-1)}\frac{(\lambda^2-v^2)(\omega-1)}{\omega}\Delta t \partial_{xx} w = O(\Delta t),$$

which can be simplified in

$$\partial_t w + v\partial_x w + \left(\frac{1}{2} - \frac{1}{\omega}\right)(\lambda^2-v^2)\Delta t \partial_{xx} w = O(\Delta t^2).$$

We can notice that we retrieve exactly the equivalent equation given in [34, 38, 50], or computed with a different formalism in the Annex A.

4 Applications for the symmetric operator $\mathcal{S}$

Using the same method, we are going to compute the equivalent systems on (w, Y) and equivalent equations on w for the symmetric operator $\mathcal{S}$ and different kinetic models. (We recall that $\mathcal{S}$ is not time-symmetric when $\omega \neq 2$, see remark 1.1. Our denomination is not necessarily the best...)

4.1 For the $D1Q2$ model

Equivalent system in $O(\Delta t^2)$

First, we consider the $D1Q2$ model. By doing a Taylor expansion of the finite difference scheme (4.2) with the Maple code given in Annex B (p.169), we obtain the equivalent system on (w, y)

$$\begin{aligned}
&\partial_t \begin{pmatrix} w \\ y \end{pmatrix} - \frac{\omega(\omega-2)(\omega^2-2\omega+2)}{2\Delta t(\omega-1)^2} \begin{pmatrix} 0 \\ y \end{pmatrix} + \begin{pmatrix} v & \gamma_1 \\ (\lambda^2-v^2)\gamma_1 & \frac{-v(\omega^4-4\omega^3+6\omega^2-4\omega+2)}{2(\omega-1)^2} \end{pmatrix} \partial_x \begin{pmatrix} w \\ y \end{pmatrix} \\
&+ \begin{pmatrix} -(\omega^2-6\omega+6)(\lambda^2-v^2) & 3v(\omega^2-2\omega+2) \\ 3v(\lambda^2-v^2)(\omega^2-2\omega+2) & -5v^2\omega^2-3\lambda^2\omega^2+6v^2\omega+10\lambda^2\omega-6v^2-10\lambda^2 \end{pmatrix} \\
&\times \frac{\Delta t\omega(\omega-2)}{32(\omega-1)^2} \partial_{xx} \begin{pmatrix} w \\ y \end{pmatrix} = O(\Delta t^2),
\end{aligned} \tag{4.6}$$

with $\gamma_1 = \frac{(\omega-2)^2(\omega^2-2\omega+2)}{8(\omega-1)^2}$.

Remark 4.1. *We can check that when we consider $\omega = 2$, we retrieve the same expression as in equation (3.9).*

Equivalent equation in $O(\Delta t^2)$

Then, we assume that we have $y = O(\Delta t)$. Let us write $y = \Delta t \tilde{y}$. We obtain

$$\frac{\omega(\omega-2)(\omega^2-2\omega+2)}{2(\omega-1)^2}\tilde{y} = \frac{(\lambda^2-v^2)(\omega-2)^2(\omega^2-2\omega+2)}{8(\omega-1)^2}\partial_x w + O(\Delta t).$$

By simplifying, we have

$$y = \frac{(\lambda^2-v^2)(\omega-2)}{4\omega}\Delta t \partial_x w + O(\Delta t^2). \tag{4.7}$$

By reinjecting this expression of y in the first equation of the equivalent system (4.6), we obtain the equivalent equation on w

$$\begin{aligned}
&\partial_t w + v\partial_x w + \frac{(\omega-2)^2(\omega^2-2\omega+2)}{8(\omega-1)^2}\frac{(\lambda^2-v^2)(\omega-2)}{4\omega}\Delta t \partial_{xx} w \\
&- \frac{\Delta t\omega(\omega-2)}{32(\omega-1)^2}(\omega^2-6\omega+6)(\lambda^2-v^2)\partial_{xx} w = O(\Delta t^2),
\end{aligned}$$

which can be simplified in

$$\partial_t w + v\partial_x w + \frac{1}{2}\left(\frac{1}{2} - \frac{1}{\omega}\right)(\lambda^2-v^2)\Delta t \partial_{xx} w = O(\Delta t^2). \tag{4.8}$$

We can notice that we recover the Dubois equivalent equation with a viscosity term divided by two, which seems reasonable because we have grouped together two time steps of the semi-symmetric operator in order to construct the symmetric operator $\mathcal{S}$.

Equivalent system and equation in $O(\Delta t^3)$

We can also compute the equivalent system and the equivalent equation of order Δt^3. We use the same methodology, but replace the approximation (4.2) of the time derivative of (w, y) by the finite difference scheme of order Δt^3

$$\partial_t \begin{pmatrix} w \\ y \end{pmatrix} (\cdot, t) = \frac{-\mathcal{S}^2(\Delta t) + 8\mathcal{S}(\Delta t) - 8\mathcal{S}^{-1}(\Delta t) + \mathcal{S}^{-2}(\Delta t)}{12\Delta t} \begin{pmatrix} w \\ y \end{pmatrix} (\cdot, t) + O(\Delta t^4), \tag{4.9}$$

Using the same Maple code than previously, we obtain an Δt^3 approximation of the form

$$\partial_t \begin{pmatrix} w \\ y \end{pmatrix} - \frac{a}{\Delta t} \begin{pmatrix} 0 \\ y \end{pmatrix} + B\partial_x \begin{pmatrix} w \\ y \end{pmatrix} + \Delta t C \partial_{xx} \begin{pmatrix} w \\ y \end{pmatrix} + \Delta t^2 D \partial_{xxx} \begin{pmatrix} w \\ y \end{pmatrix} = O(\Delta t^3). \tag{4.10}$$

The value of a and the coefficients of the matrices B, C and D will be replaced later by their values given in Maple. Now, we can assume that $y = O(\Delta t)$, and that $y = \Delta t \tilde{y}$. We obtain that

$$a\tilde{y} = b_{2,1}\partial_x w + O(\Delta t),$$

which can be written as

$$y = \Delta t \frac{b_{2,1}}{a} \partial_x w + O(\Delta t^2). \tag{4.11}$$

We also have

$$a\tilde{y} = b_{2,1}\partial_x w + \Delta t \left(\partial_t \tilde{y} + b_{2,2}\partial_x \tilde{y} + c_{2,1}\partial_{xx} w\right) + O(\Delta t^2),$$

which can be written as

$$y = \frac{\Delta t}{a} \left(b_{2,1}\partial_x w + \partial_t y + b_{2,2}\partial_x y\right) + \frac{\Delta t^2}{a} c_{2,1}\partial_{xx} w + O(\Delta t^3). \tag{4.12}$$

By deriving in time the equation (4.11), and assuming that it does not break the order hierarchy in Δt we have

$$\partial_t y = \Delta t \frac{b_{2,1}}{a} \partial_{tx} w + O(\Delta t^2).$$

Using the first equation of (4.10), we have

$$\begin{aligned} \partial_t y &= \Delta t \frac{b_{2,1}}{a} \partial_x \left(-b_{1,1}\partial_x w + O(\Delta t)\right) + O(\Delta t^2), \\ &= -\Delta t \frac{b_{2,1} b_{1,1}}{a} \partial_{xx} w + O(\Delta t^2). \end{aligned}$$

Moreover, the derivation with respect to x of (4.11) gives us

$$\partial_x y = \Delta t \frac{b_{2,1}}{a} \partial_{xx} w + O(\Delta t^2).$$

By replacing in (4.12) the expression of $\partial_t y$ and $\partial_x y$, we obtain an expression of y in function of the space derivatives of w

$$y = \frac{\Delta t}{a} b_{2,1}\partial_x w + \Delta t^2 \left(\frac{c_{2,1}}{a} - \frac{b_{2,1} b_{1,1}}{a^2} + \frac{b_{2,2} b_{2,1}}{a^2}\right) \partial_{xx} w + O(\Delta t^3). \tag{4.13}$$

By reinjecting this expression of y in the first equation of the equivalent system on (w, y) (4.10), we obtain

$$\begin{aligned}\partial_t w + b_{1,1}\partial_x w + b_{1,2}\partial_x \left[\frac{\Delta t}{a} b_{2,1}\partial_x w + \Delta t^2 \left(\frac{c_{2,1}}{a} - \frac{b_{2,1}b_{1,1}}{a^2} + \frac{b_{2,2}b_{2,1}}{a^2}\right)\partial_{xx} w\right] \\ + \Delta t c_{1,1}\partial_{xx} w + \Delta t c_{1,2}\partial_{xx}\left[\Delta t \frac{b_{2,1}}{a}\partial_x w\right] + \Delta t^2 d_{1,1}\partial_{xxx} w = O(\Delta t^3).\end{aligned}$$

This can be simplified in

$$\begin{aligned}\partial_t w + b_{1,1}\partial_x w + \Delta t \left(\frac{b_{1,2}b_{2,1}}{a} + c_{1,1}\right)\partial_{xx} w \\ + \Delta t^2 \left(c_{1,2}\frac{b_{2,1}}{a} + b_{1,2}\left(\frac{c_{2,1}}{a} - \frac{b_{2,1}b_{1,1}}{a^2} + \frac{b_{2,2}b_{2,1}}{a^2}\right) + d_{1,1}\right)\partial_{xxx} w = O(\Delta t^3).\end{aligned}$$

By replacing the coefficients by their values given in Maple, we obtain the equivalent equation

$$\partial_t w + v\partial_x w + \frac{\Delta t}{2}\left(\frac{1}{2} - \frac{1}{\omega}\right)(\lambda^2 - v^2)\partial_{xx} w + \Delta t^2 \frac{\omega^2 - 6\omega + 6}{12\omega^2}(v^2 - \lambda^2) v \partial_{xxx} w = O(\Delta t^3). \tag{4.14}$$

Remark 4.2. *We can notice that when the relaxation parameter $\omega = 3 - \sqrt{3} \approx 1,2679$, the dispersive term in $O(\Delta t^2)$ vanishes.*

4.2 For the $D2Q3$ model

Equivalent system on (w, Y)

We can also compute the equivalent system of the $D2Q3$ model, using the Maple code given in Annex B (p.169). We obtain

$$\begin{aligned}\partial_t \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} - \frac{\omega(\omega-2)(\omega^2 - 2\omega + 2)}{2\Delta t(\omega - 1)^2}\begin{pmatrix} 0 \\ y_1 \\ y_2 \end{pmatrix} \\ + \begin{pmatrix} v_1 & -2\gamma_1 & 0 \\ \gamma_1(2v_1 + \lambda)(v_1 - \lambda) & \gamma_2(2v_1 - \lambda) & 0 \\ \gamma_1 v_2(2v_1 + \lambda) & 2v_2\gamma_2 & \gamma_2\lambda \end{pmatrix}\partial_1 \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} \\ + \begin{pmatrix} v_2 & 0 & -2\gamma_1 \\ \gamma_1 v_2(2v_1 + \lambda) & 0 & \gamma_2(2v_1 + \lambda) \\ \gamma_1(v_1\lambda + 2{v_2}^2 - \lambda^2) & \gamma_2\lambda & 2v_2\gamma_2 \end{pmatrix}\partial_2 \begin{pmatrix} w \\ y_1 \\ y_2 \end{pmatrix} = O(\Delta t),\end{aligned} \tag{4.15}$$

with $\gamma_1 = -\frac{1}{16}\frac{(\omega^2-2\omega+2)(\omega-2)^2}{(\omega-1)^2}$ and $\gamma_2 = -\frac{1}{4}\frac{\omega^4-4\omega^3+6\omega^2-4\omega+2}{(\omega-1)^2}$.

Remark 4.3. *When $\omega = 2$, the damping term vanishes, $\gamma_1 = 0$ and $\gamma_2 = \frac{1}{2}$, which allows us to retrieve the equivalent system (3.10), obtained in Chapter 3. We can notice that, when $\omega = 2$, the equation on w is independent of the system on Y, but it is not the case anymore when $\omega < 2$.*

Equivalent equation on w

Now, let us assume that $Y = O(\Delta t)$. We obtain

$$y_1 = \Delta t \left(\frac{1}{\omega} - \frac{1}{2}\right) \frac{(2v_1 + \lambda)}{2} \left((v_1 - \lambda)\, \partial_1 w + v_2 \partial_2 w\right) + O(\Delta t^2),$$

and

$$y_2 = \frac{\Delta t}{2} \left(\frac{1}{\omega} - \frac{1}{2}\right) \left((2v_2 v_1 + v_2 \lambda)\, \partial_1 w + \left(\lambda v_1 + 2{v_2}^2 - \lambda^2\right) \partial_2 w\right) + O(\Delta t^2).$$

By reinjecting these expressions of y_1 and y_2 in the first equation of the equivalent system (4.15), we obtain

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = \frac{\Delta t}{2} \left(\frac{1}{\omega} - \frac{1}{2}\right) \nabla \cdot (\mathcal{D}_3 \nabla w) + O(\Delta t^2), \tag{4.16}$$

with the diffusion matrix

$$\mathcal{D}_3 = \begin{pmatrix} \frac{\lambda}{2}(\lambda + v_1) - {v_1}^2 & -\frac{\lambda}{2} v_2 - v_1 v_2 \\ -\frac{\lambda}{2} v_2 - v_1 v_2 & \frac{\lambda}{2}(\lambda - v_1) - {v_2}^2 \end{pmatrix}.$$

Remark 4.4. *We retrieve the same equivalent equation as the one given by the computation in the Annex A, with a diffusion term divided by two, which seems reasonable because we have grouped together two time steps of the semi-symmetric operator in order to construct the symmetric operator $\mathcal{S}$.*

4.3 For the $D2Q4$ model

Equivalent system on (w, Y)

We can also compute the equivalent system on (w, Y) of the $D2Q4$ model, using the Maple code given in Annex B (p.169). We obtain

$$\begin{aligned} \partial_t \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} - \frac{\omega(\omega-2)(\omega^2 - 2\omega + 2)}{2\Delta t(\omega - 1)^2} \begin{pmatrix} 0 \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} + \begin{pmatrix} v_1 & 2\gamma_1 & 0 & 0 \\ \gamma_1(\lambda^2 - 2{v_1}^2) & -2v_1\gamma_2 & 0 & \gamma_2 \\ -2v_1 v_2 \gamma_1 & -2v_2\gamma_2 & 0 & 0 \\ 2\lambda^2 v_1 \gamma_1 & 2\lambda^2 \gamma_2 & 0 & 0 \end{pmatrix} \partial_1 \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} \\ + \begin{pmatrix} v_2 & 0 & 2\gamma_1 & 0 \\ -2v_1 v_2 \gamma_1 & 0 & -2v_1\gamma_2 & 0 \\ \gamma_1(\lambda^2 - 2{v_2}^2) & 0 & -2v_2\gamma_2 & -\gamma_2 \\ -2\lambda^2 v_2 \gamma_1 & 0 & -2\lambda^2\gamma_2 & 0 \end{pmatrix} \partial_2 \begin{pmatrix} w \\ y_1 \\ y_2 \\ z_3 \end{pmatrix} = O(\Delta t), \end{aligned} \tag{4.17}$$

with $\gamma_1 = \frac{(\omega-2)^2(\omega^2-2\omega+2)}{16(\omega-1)^2}$ and $\gamma_2 = \frac{\omega^4-4\omega^3+6\omega^2-4\omega+2}{4(\omega-1)^2}$.

Remark 4.5. *When $\omega = 2$, the damping term vanishes, $\gamma_1 = 0$ and $\gamma_2 = \frac{1}{2}$, which allows us to retrieve the equivalent system (3.11), obtained in Chapter 3.*

Equivalent equation on w

As previously, we assume that $Y = O(\Delta t)$. We obtain the equivalent equation

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = \frac{\Delta t}{2}\left(\frac{1}{\omega} - \frac{1}{2}\right)\nabla \cdot (\mathcal{D}_4 \nabla w) + O(\Delta t^2),$$

with the diffusion matrix

$$\mathcal{D}_4 = \begin{pmatrix} \frac{\lambda^2}{2} - {v_1}^2 & -v_1 v_2 \\ -v_1 v_2 & \frac{\lambda^2}{2} - {v_2}^2 \end{pmatrix}.$$

Remark 4.6. *We retrieve the same equivalent equation as the one given by the computation in the Annex A, with a diffusion term divided by two, which seems reasonable because we have grouped together two time steps of the semi-symmetric operator in order to construct the symmetric operator $\mathcal{S}$.*

In conclusion of this section, we have computed the equivalent equations for all the kinetic models introduced in Chapter 2. For each model, we have obtained the equivalent system on (w, Y) and the equivalent equation on w. We shall see in Section 2 that the two equivalent equations do not have the same stability properties and that the equivalent system on (w, Y) gives more precise information about the stability of the model. In [14], Bouchut reviews several stability conditions for kinetic models. According to our analysis, it seems that the stability condition that we find for the equivalent system on (w, Y) is as strong as the Entropy Extension Condition (EEC) of Bouchut [14]. A possible advantage of our approach is its purely algebraic nature, which makes it easy to compute.

5 Comparison of the equivalent system on (w, Y) and the equivalent equation on w for the $D1Q2$ model

Now that we have obtained the equivalent equations, we wish to quantify numerically how they are close to the kinetic equations. We shall compute analytic solutions of the equivalent equations and compare them with the solutions of the kinetic equation. We shall also compute the error between the two solutions.

Let us consider a particular solution of the form

$$\begin{pmatrix} w \\ y \end{pmatrix}(x,t) = \begin{pmatrix} w_0 \\ y_0 \end{pmatrix} e^{\alpha t} e^{ikx}, \tag{4.18}$$

with $k \in \mathbb{N}$ and $\alpha \in \mathbb{C}$.

5.1 Particular solution of the equivalent equation

If we inject this particular solution (4.18) in the equivalent equation (4.8) on w, we obtain

$$\alpha w + ivkw = -\frac{\Delta t}{2}\left(\frac{1}{\omega} - \frac{1}{2}\right) k^2 \left(\lambda^2 - v^2\right) w.$$

It gives us the value of α with respect to k and v

$$\alpha = -\frac{\Delta t}{2}\left(\frac{1}{\omega}-\frac{1}{2}\right)k^2\left(\lambda^2-v^2\right)-ivk.$$

A particular solution of the equivalent equation (4.8) is then

$$w = w_0 e^{-\left(\frac{\Delta t}{2}\left(\frac{1}{\omega}-\frac{1}{2}\right)k^2\left(\lambda^2-v^2\right)+ivk\right)t}e^{ikx}.$$

In order to deal with real solutions, we compute the real part of this particular solution, that we denote w_{EqEq} and which is still a solution of (4.8)

$$w_{\text{EqEq}} = \Re(w) = w_0 e^{-\frac{\Delta t}{2}\left(\frac{1}{\omega}-\frac{1}{2}\right)k^2\left(\lambda^2-v^2\right)t}\cos\left(k(x-vt)\right).$$

To compute the equivalent equation, we assume that we have the relation between y and $\partial_x w$ given by (4.7)

$$y = \frac{(\lambda^2-v^2)(\omega-2)}{4\omega}\Delta t\partial_x w. \tag{4.19}$$

We denote y_{EqEq} the real part of y

$$\begin{aligned} y_{\text{EqEq}} &= \Re(y), \\ &= -\frac{(\lambda^2-v^2)(\omega-2)}{4\omega}\Delta t k w_0 e^{-\frac{\Delta t}{2}\left(\frac{1}{\omega}-\frac{1}{2}\right)k^2\left(\lambda^2-v^2\right)t}\sin\left(k(x-vt)\right). \end{aligned}$$

5.2 Particular solution of the equivalent system

Now, we inject the expression of the particular solution (4.18) in the equivalent system (4.6). We obtain

$$\left(\alpha I_2 - \frac{1}{\Delta t}A + iBk - \Delta t C k^2\right)\begin{pmatrix} w \\ y \end{pmatrix} = 0,$$

with $A = \begin{pmatrix} 0 & 0 \\ 0 & a \end{pmatrix}$.

The previous system admits two solutions $\alpha_1(k)$ and $\alpha_2(k)$ depending on k, which are the eigenvalues of $\frac{1}{\Delta t}A - iBk + \Delta t C k^2$. We have

$$\alpha_1(k) = \frac{1}{\Delta t}\frac{16\omega^4-64\omega^3+96\omega^2-64\omega}{32(\omega-1)^2} + O(\Delta t^0),$$

and

$$\alpha_2(k) = ikc - \frac{c^2\lambda^2\omega^2-2c^2+2\lambda^2}{4\omega}k^2\Delta t + O(\Delta t^2).$$

One of the solutions, $\alpha_1(k)$, behaves as $O(\frac{1}{\Delta t})$ when $\Delta t \to 0$, and the real part of the other solution $\alpha_2(k)$ behaves as $O(\Delta t)$ when $\Delta t \to 0$. If we compute the particular solution (4.18) with the eigenvalue α_1 in $O(\frac{1}{\Delta t})$, we observe that y decreases rapidly toward 0.

If we consider instead, the solution given by the second eigenvalue α_2, y stays small and has slower variations. We choose to keep this eigenvalue α_2 for a relevant comparison with the expected behavior.

A particular solution of the equivalent system (4.6) is then

$$\begin{pmatrix} w \\ y \end{pmatrix}(x,t) = \begin{pmatrix} w_0 \\ y_0 \end{pmatrix} e^{\alpha_2 t} e^{ikx}.$$

To test, we compute the real part of w, that we denote w_{EqSys}

$$w_{\text{EqSys}} = \Re(w) = w_0 e^{\Re(\alpha_2)t} \cos\left(\Im(\alpha_2)t + kx\right),$$

and the real part of y, denoted by y_{EqSys}

$$y_{\text{EqSys}} = \Re(y) = e^{\Re(\alpha_2)t}\left(\Re(y_0)\cos\left(\Im(\alpha_2)t + kx\right) - \Im(y_0)\sin\left(\Im(\alpha_2)t + kx\right)\right).$$

5.3 Numerical comparison of w

We take $k = 2$. We choose $w_0 = 1$, and we take y_0 such as (w_0, y_0) belong to the kernel of the matrix $\frac{1}{\Delta t}A - iBk + \Delta t C k^2$.

We denote w_{LB} the solution given by the $D1Q2$ model with the initialization

$$\begin{pmatrix} w_{\text{LB}} \\ y_{\text{LB}} \end{pmatrix}(x,0) = \begin{pmatrix} w_0 \\ y_0 \end{pmatrix} \cos(kx).$$

We compute the relative L^2 error between the solution of the equivalent equation w_{EqEq} and the solution given by the $D1Q2$ model w_{LB} at the final time

$$\sqrt{\frac{\sum_{i=0}^{Nx}\left(w_{\text{LB}}^{i,Nt} - w_{\text{EqEq}}^{i,Nt}\right)^2}{\sum_{i=0}^{Nx}\left(w_{\text{LB}}^{i,Nt}\right)^2}},$$

and the relative L^2 error between the solution of the equivalent system w_{EqSys} and w_{LB}

$$\sqrt{\frac{\sum_{i=0}^{Nx}\left(w_{\text{LB}}^{i,Nt} - w_{\text{EqSys}}^{i,Nt}\right)^2}{\sum_{i=0}^{Nx}\left(w_{\text{LB}}^{i,Nt}\right)^2}}.$$

We take $T = \pi$. We compute the solution for different amounts of time steps $Nt = 16, 32, 64, 128, 256, 512, 1024$ and 2048, which gives us different time steps $\Delta t = \frac{T}{Nt}$.

We obtain the relative errors of Figure 4.1, for different relaxation parameters ω.

The equivalent equation and the equivalent system both converge at the order 2 toward the solution given by the $D1Q2$ model. When $\omega \in [1.8, 2]$, the equivalent equation and the equivalent system give similar accuracy. When $\omega \in [1.5, 1.8]$, the equivalent system is a better approximation of the solution given by the $D1Q2$ model, while when $\omega \leqslant 1.4$, the equivalent equation is more accurate.

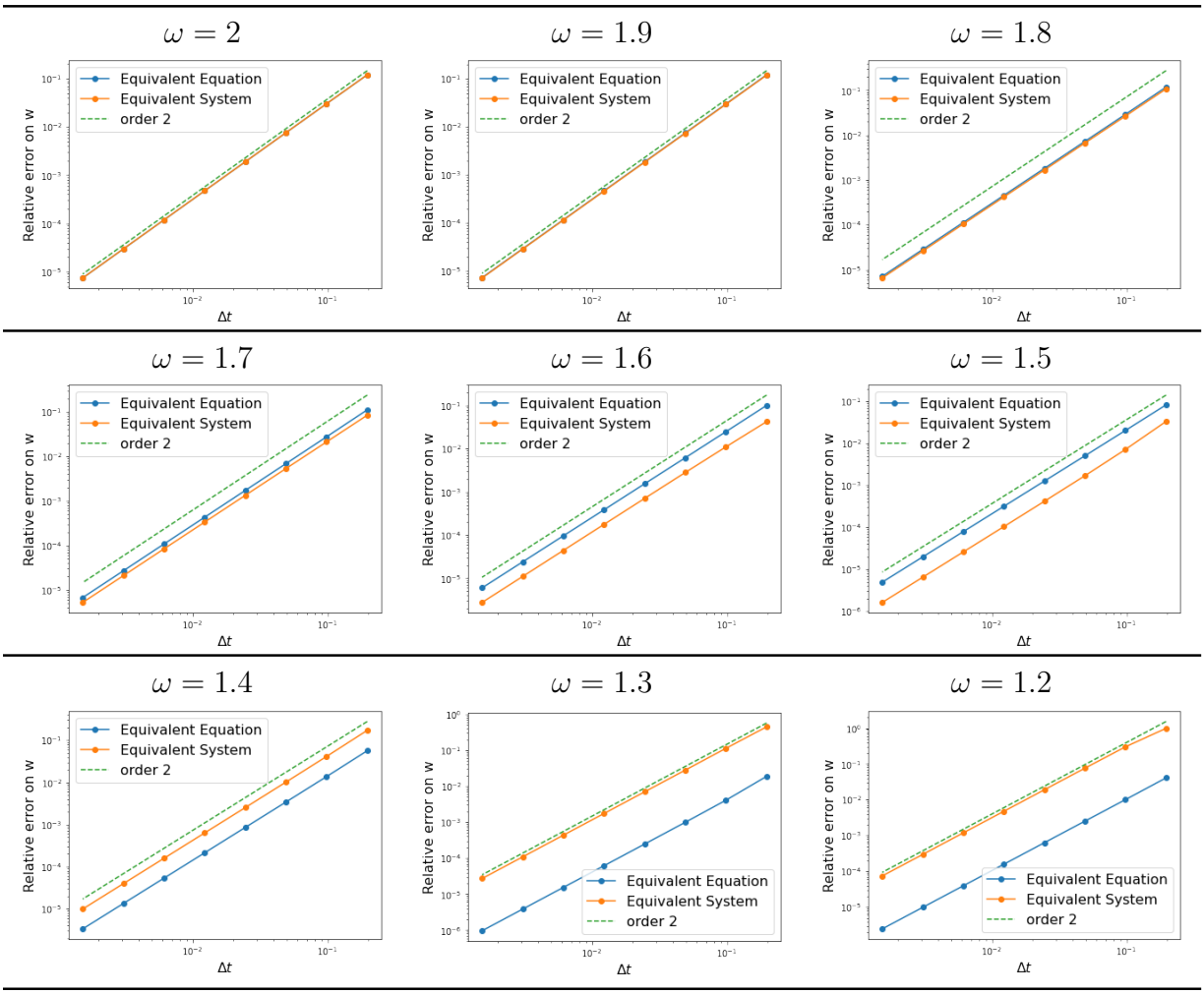

Table 4.1: Relative L^2 error on w with respect to the time step Δt, for different relaxation parameters ω.

5.4 Numerical comparison of y

Now, we want to compute the error on the flux error y.

The flux error of the $D1Q2$ model y_{LB} is given by

$$y_{\text{LB}} = \sum_{i=1}^{2} \lambda_i f_i - v \left(\sum_{i=1}^{2} f_i \right).$$

We can compute the relative L^2 errors between y_{LB} and the flux error y_{EqEq} that we assume to have in order to compute the equivalent equation and between y_{LB} and the solution of

the equivalent system y_{EqSys}

$$\sqrt{\frac{\sum_{i=0}^{Nx}\left(y_{\mathrm{LB}}^{i,Nt}-y_{\mathrm{EqEq}}^{i,Nt}\right)^2}{\sum_{i=0}^{Nx}\left(y_{\mathrm{LB}}^{i,Nt}\right)^2}} \quad \text{and} \quad \sqrt{\frac{\sum_{i=0}^{Nx}\left(y_{\mathrm{LB}}^{i,Nt}-y_{\mathrm{EqSys}}^{i,Nt}\right)^2}{\sum_{i=0}^{Nx}\left(y_{\mathrm{LB}}^{i,Nt}\right)^2}}.$$

We obtain the Figure 4.2. We can observe that the flux error y_{EqEq} given by the equivalent equation converges at the order 1 toward the y_{LB} given by the $D1Q2$ model, while the y_{EqSys} given by the equivalent system converges at the order 2.

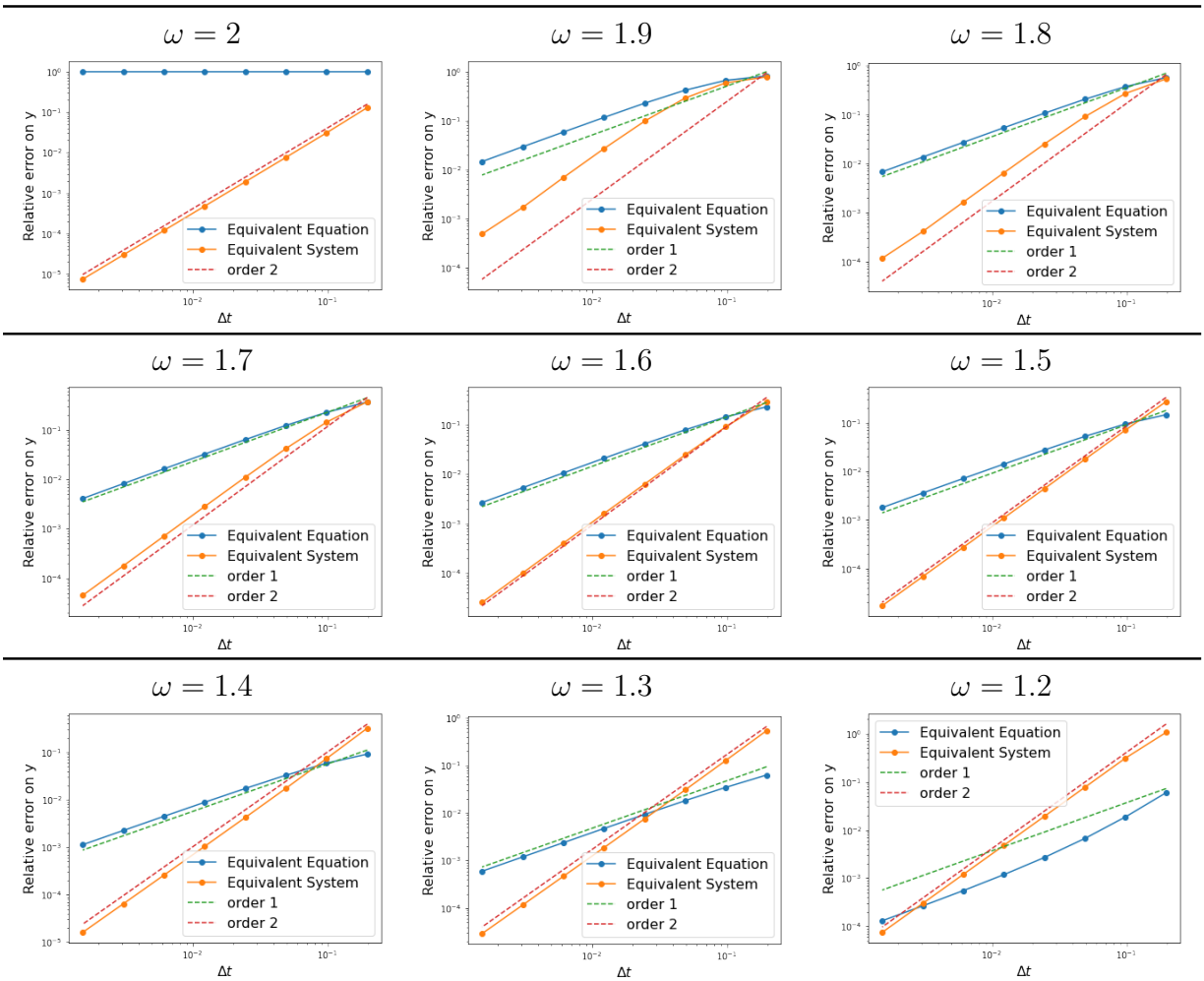

Table 4.2: Relative L^2 error on y with respect to the time step Δt, for different relaxation parameters ω.

Remark 5.1. *When $\omega = 2$, the error between the flux error y given by the equivalent equation and the one given by the $D1Q2$ model is constant. This is due to the fact that y_{EqEq} is given by (4.19), which is equal to 0 when $\omega = 2$. Indeed, when $\omega = 2$, w and y are independent, so we do not have to assume the smallness hypothesis $y = O(\Delta t)$ to deduce the equivalent equation from the equivalent system.*

6 Conclusion

To conclude, in this chapter, we have proposed a generalization of the equivalent system on (w, Y) presented in Chapter 3 for the case of a relaxation parameter $\omega < 2$. We have shown that from this equivalent system, we can retrieve the classical equivalent equation on w proposed in [34, 38, 50] by assuming that $Y = O(\Delta t)$.

We have numerically compared the equivalent system on (w, y) and the equivalent equation on w in the case of the $D1Q2$ model. We obtain that the equivalent system is a better approximation of the solution w given by the kinetic model when ω is close to 2, while the equivalent equation is better when ω is close to 1. For all the values of ω, the equivalent system gives more precise information on the behavior of the flux error y than the equivalent equation.

Chapter 5

Two-scale convergence

In this chapter, we propose a tentative analysis of the equivalent system by a two-scale convergence analysis. The two-scale convergence is a general tool, proposed by Allaire in [1] (see also the presentation given by Frénod in [47]), for analyzing PDEs with small parameters generating fast oscillations. The two-scale analysis transforms the initial PDE with one time variable t and a small parameter ε into a PDE with two time variables t and τ and no small parameter. The new PDE gives insights on how the slow scale and fast scale separates when the small parameter ε goes to zero. We will see that, unfortunately, our analysis is not completely conclusive. At the end, we will try to give some possible ways to improve the analysis.

1 Two-scale convergence

Definition 1.1. *A sequence* $\begin{pmatrix} w_{\Delta t} \\ Y_{\Delta t} \end{pmatrix}$ *two-scale converges to* $\boldsymbol{U} = \begin{pmatrix} w \\ Y \end{pmatrix}$ *if for all function* $\boldsymbol{\varphi} \in \mathcal{C}^1(\mathbb{R}^2; \mathcal{C}^1([0,T); \mathcal{C}^1([0,\tau_{\max}])))$*, we have*

$$\int_0^T \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} \cdot \boldsymbol{\varphi}\left(\boldsymbol{x}, t, \frac{t}{\Delta t}\right) dt \xrightarrow[\Delta t \to 0]{} \int_0^T \int_0^{\tau_{\max}} \boldsymbol{U}(\boldsymbol{x},t,\tau) \cdot \boldsymbol{\varphi}(\boldsymbol{x},t,\tau) dt d\tau.$$

Theorem 1.1. *If the sequence* $\begin{pmatrix} w_{\Delta t} \\ Y_{\Delta t} \end{pmatrix}$ *satisfies the equivalent equation of the D2Q3 model* (4.15) *and two-scale converges toward* $\boldsymbol{U}$*, then there exists a vectorial function* $\boldsymbol{\mu}$ *such that*

$$\boldsymbol{U}(\boldsymbol{x},t,\tau) = \begin{pmatrix} \mu_0(\boldsymbol{x},t) \\ \mu_1(\boldsymbol{x},t)e^{a\tau} \\ \mu_2(\boldsymbol{x},t)e^{a\tau} \end{pmatrix}, \tag{5.1}$$

and $\boldsymbol{U}$ *satisfies the equation*

$$\partial_t \boldsymbol{U}(\boldsymbol{x},t,\tau) + \begin{pmatrix} v_1 & -2\gamma_1(\tau) & 0 \\ \gamma_1(\tau)(2v_1+\lambda)(v_1-\lambda) & \gamma_2(\tau)(2v_1-\lambda) & 0 \\ \gamma_1(\tau)v_2(2v_1+\lambda) & 2v_2\gamma_2(\tau) & \gamma_2(\tau)\lambda \end{pmatrix} \partial_1 \boldsymbol{U}(\boldsymbol{x},t,\tau)$$
$$+ \begin{pmatrix} v_2 & 0 & -2\gamma_1(\tau) \\ \gamma_1(\tau)v_2(2v_1+\lambda) & 0 & \gamma_2(\tau)(2v_1+\lambda) \\ \gamma_1(\tau)(v_1\lambda + 2{v_2}^2 - \lambda^2) & \gamma_2(\tau)\lambda & 2v_2\gamma_2(\tau) \end{pmatrix} \partial_2 \boldsymbol{U}(\boldsymbol{x},t,\tau) = O(\Delta t), \tag{5.2}$$

with

$$\gamma_1(\tau) = -\frac{(\omega^2-2\omega+2)(\omega-2)^2}{16(\omega-1)^2}\frac{e^{a\tau}}{\tau_{\max}}\left(\frac{e^{a\tau_{\max}}-1}{a}\right),$$
$$\gamma_2(\tau) = \frac{(\omega^4-4\omega^3+6\omega^2-4\omega+2)}{4(\omega-1)^2}\frac{e^{-a\tau}}{\tau_{\max}}\left(\frac{e^{-a\tau_{\max}}-1}{a}\right),$$

and

$$a = \frac{\omega(\omega-2)(\omega^2-2\omega+2)}{2\Delta t(\omega-1)^2}$$

Remark 1.1. *The previous theorem gives the two-scale convergence of the solution obtained with the D2Q3 model, but we can achieve a similar theorem with the D2Q4 model.*

Proof. As $\begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix}$ satisfy (4.15), we have

$$\partial_t \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} - \frac{a}{\Delta t}\begin{pmatrix} 0 \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} + B_1\partial_1 \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} + B_2\partial_2 \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} = O(\Delta t). \tag{5.3}$$

By multiplying by a test function $\boldsymbol{\varphi}\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right) \in \mathcal{C}^1(\mathbb{R}^2;\mathcal{C}_0^1([0,T);\mathcal{C}^1([0,\tau_{\max}])))$ and integrating, we obtain

$$\int_0^T \int_{\mathbb{R}^2} \left(\partial_t \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} - \frac{a}{\Delta t}\begin{pmatrix} 0 \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} + \sum_{k=1}^2 B_k\partial_k \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} \right)$$
$$\cdot \boldsymbol{\varphi}\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right) d\boldsymbol{x}dt = 0.$$

By integrating by parts the previous equation, we obtain

$$\int_0^T \int_{\mathbb{R}^2} \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},t) \\ Y_{\Delta t}(\boldsymbol{x},t) \end{pmatrix} \cdot \left(\partial_t \boldsymbol{\varphi}\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right) + \frac{1}{\Delta t}\partial_\tau \boldsymbol{\varphi}(\boldsymbol{x},t,\tau) + \frac{a}{\Delta t}\begin{pmatrix} 0 \\ \varphi_1\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right) \\ \varphi_2\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right) \end{pmatrix} \right.$$
$$\left. + \sum_{k=1}^2 B_k\partial_k \boldsymbol{\varphi}\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right) \right) d\boldsymbol{x}dt \tag{5.4}$$
$$+ \int_{\mathbb{R}^2} \left(\begin{pmatrix} w_{\Delta t}(\boldsymbol{x},0) \\ Y_{\Delta t}(\boldsymbol{x},0) \end{pmatrix} \cdot \boldsymbol{\varphi}(\boldsymbol{x},0,0) - \begin{pmatrix} w_{\Delta t}(\boldsymbol{x},T) \\ Y_{\Delta t}(\boldsymbol{x},T) \end{pmatrix} \cdot \boldsymbol{\varphi}\left(\boldsymbol{x},T,\frac{T}{\Delta t}\right) \right) d\boldsymbol{x} = 0.$$

As $\boldsymbol{\varphi}$ vanishes at $t=0$ and $t=T$, we have

$$\begin{aligned}&\int_0^T\int_{\mathbb{R}^2}\begin{pmatrix}w_{\Delta t}(\boldsymbol{x},t)\\ Y_{\Delta t}(\boldsymbol{x},t)\end{pmatrix}\cdot\left(\partial_t\boldsymbol{\varphi}\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right)+\frac{1}{\Delta t}\partial_\tau\boldsymbol{\varphi}\left(\boldsymbol{x},t,\tau\right)+\frac{a}{\Delta t}\begin{pmatrix}0\\ \varphi_1\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right)\\ \varphi_2\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right)\end{pmatrix}\right.\\ &\left.+\sum_{k=1}^{2}B_k\partial_k\boldsymbol{\varphi}\left(\boldsymbol{x},t,\frac{t}{\Delta t}\right)\right)d\boldsymbol{x}dt=0.\end{aligned}\tag{5.5}$$

If we multiply the previous equation by Δt and take the two-scale limit, we obtain

$$\int_0^T\int_0^{\tau_{\max}}\int_{\mathbb{R}^2}\boldsymbol{U}(\boldsymbol{x},t,\tau)\cdot\left(\partial_\tau\boldsymbol{\varphi}\left(\boldsymbol{x},t,\tau\right)+a\begin{pmatrix}0\\ \varphi_1\left(\boldsymbol{x},t,\tau\right)\\ \varphi_2\left(\boldsymbol{x},t,\tau\right)\end{pmatrix}\right)d\boldsymbol{x}d\tau dt=0.\tag{5.6}$$

By integrating by parts, we obtain

$$\begin{aligned}&\int_0^T\int_0^{\tau_{\max}}\int_{\mathbb{R}^2}\left(\partial_\tau\boldsymbol{U}\left(\boldsymbol{x},t,\tau\right)-a\begin{pmatrix}0\\ U_1\left(\boldsymbol{x},t,\tau\right)\\ U_2\left(\boldsymbol{x},t,\tau\right)\end{pmatrix}\right)\cdot\boldsymbol{\varphi}\left(\boldsymbol{x},t,\tau\right)d\boldsymbol{x}d\tau dt\\ &+\int_0^T\int_{\mathbb{R}^2}\left(\boldsymbol{U}(\boldsymbol{x},t,0)\boldsymbol{\varphi}(\boldsymbol{x},t,0)-\boldsymbol{U}(\boldsymbol{x},t,\tau_{\max})\boldsymbol{\varphi}(\boldsymbol{x},t,\tau_{\max})\right)d\boldsymbol{x}dt=0.\end{aligned}\tag{5.7}$$

Let us impose that the test functions $\boldsymbol{\varphi}$ satisfy the condition

$$\boldsymbol{U}(\boldsymbol{x},t,0)\boldsymbol{\varphi}(\boldsymbol{x},t,0)-\boldsymbol{U}(\boldsymbol{x},t,\tau_{\max})\boldsymbol{\varphi}(\boldsymbol{x},t,\tau_{\max})=0.\tag{5.8}$$

Then we have the equation

$$\partial_\tau\boldsymbol{U}(\boldsymbol{x},t,\tau)=a\begin{pmatrix}0\\ U_1(\boldsymbol{x},t,\tau)\\ U_2(\boldsymbol{x},t,\tau)\end{pmatrix}.\tag{5.9}$$

The condition (5.9) leads to:

- As we have $\partial_\tau U_0=0$, U_0 does not depend on the variable τ.
- For $i=1,2$, we have $\partial_\tau U_i=aU_i$. So we have a solution of the form $U_i=U_i(t,0)e^{a\tau}$.

Finally, there exists $\boldsymbol{\mu}$ such as

$$\boldsymbol{U}(\boldsymbol{x},t,\tau)=\begin{pmatrix}\mu_0(\boldsymbol{x},t)\\ \mu_1(\boldsymbol{x},t)e^{a\tau}\\ \mu_2(\boldsymbol{x},t)e^{a\tau}\end{pmatrix}=D(\tau)\boldsymbol{\mu}(\boldsymbol{x},t),\tag{5.10}$$

with the matrix

$$D(\tau)=\begin{pmatrix}1&0&0\\ 0&e^{a\tau}&0\\ 0&0&e^{a\tau}\end{pmatrix}.\tag{5.11}$$

We now choose a test function $\boldsymbol{\varphi}$ which satisfies

$$\partial_\tau \boldsymbol{\varphi}(\boldsymbol{x}, t, \tau) = -a \begin{pmatrix} 0 \\ \varphi_1(\boldsymbol{x}, t, \tau) \\ \varphi_2(\boldsymbol{x}, t, \tau) \end{pmatrix}. \tag{5.12}$$

Then, there exists $\boldsymbol{\theta}$ such as

$$\boldsymbol{\varphi}(\boldsymbol{x}, t, \tau) = \begin{pmatrix} \theta_0(\boldsymbol{x}, t) \\ \theta_1(\boldsymbol{x}, t) e^{-a\tau} \\ \theta_2(\boldsymbol{x}, t) e^{-a\tau} \end{pmatrix} = D(-\tau)\boldsymbol{\theta}(\boldsymbol{x}, t). \tag{5.13}$$

Let us remark, that the condition (5.8) is satisfied with a test function $\boldsymbol{\varphi}$ of this form. Indeed, we have

$$\begin{aligned} &\boldsymbol{U}(\boldsymbol{x}, t, 0) \cdot \boldsymbol{\varphi}(\boldsymbol{x}, t, 0) - \boldsymbol{U}(\boldsymbol{x}, t, \tau_{\max}) \cdot \boldsymbol{\varphi}(\boldsymbol{x}, t, \tau_{\max}) \\ &\qquad = D(0)\boldsymbol{\mu}(\boldsymbol{x}, t) \cdot D(0)\boldsymbol{\theta}(\boldsymbol{x}, t) - D(\tau_{\max})\boldsymbol{\mu}(\boldsymbol{x}, t) \cdot D(-\tau_{\max})\boldsymbol{\theta}(\boldsymbol{x}, t), \\ &\qquad = \boldsymbol{\mu}(\boldsymbol{x}, t) \cdot \boldsymbol{\theta}(\boldsymbol{x}, t) - \boldsymbol{\mu}(\boldsymbol{x}, t) \cdot \boldsymbol{\theta}(\boldsymbol{x}, t), \\ &\qquad = \boldsymbol{0}. \end{aligned}$$

Thereby, the equation (5.5) gives us

$$\int_0^T \int_{\mathbb{R}^2} \begin{pmatrix} w_{\Delta t}(\boldsymbol{x}, t) \\ Y_{\Delta t}(\boldsymbol{x}, t) \end{pmatrix} \cdot \left(\partial_t \boldsymbol{\varphi}\left(\boldsymbol{x}, t, \frac{t}{\Delta t}\right) + \sum_{k=1}^{2} B_k \partial_k \boldsymbol{\varphi}\left(\boldsymbol{x}, t, \frac{t}{\Delta t}\right) \right) d\boldsymbol{x} dt = 0.$$

By taking the two-scale limit, we have

$$\int_0^T \int_0^{\tau_{\max}} \int_{\mathbb{R}^2} \boldsymbol{U}(\boldsymbol{x}, t, \tau) \cdot \left(\partial_t \boldsymbol{\varphi}(\boldsymbol{x}, t, \tau) + \sum_{k=1}^{2} B_k \partial_k \boldsymbol{\varphi}(\boldsymbol{x}, t, \tau) \right) d\boldsymbol{x} d\tau dt = 0.$$

Then, replacing $\boldsymbol{U}$ and $\boldsymbol{\varphi}$ respectively by (5.10) and (5.13), we obtain

$$\int_0^T \int_0^{\tau_{\max}} \int_{\mathbb{R}^2} (D(\tau)\boldsymbol{\mu}(\boldsymbol{x}, t)) \cdot \left(D(-\tau)\partial_t \boldsymbol{\theta}(\boldsymbol{x}, t) + \sum_{k=1}^{2} B_k D(-\tau) \partial_k \boldsymbol{\theta}(\boldsymbol{x}, t) \right) d\boldsymbol{x} d\tau dt = 0.$$

As B is diagonal, we have

$$\int_0^T \int_0^{\tau_{\max}} \int_{\mathbb{R}^2} \boldsymbol{\mu}(\boldsymbol{x}, t) \cdot \left(\partial_t \boldsymbol{\theta}(\boldsymbol{x}, t) + \sum_{k=1}^{2} D(\tau) B_k D(-\tau) \partial_k \boldsymbol{\theta}(\boldsymbol{x}, t) \right) d\boldsymbol{x} d\tau dt = 0.$$

Therefore, we have

$$\int_0^T \int_{\mathbb{R}^2} \boldsymbol{\mu}(\boldsymbol{x}, t) \cdot \left(\partial_t \boldsymbol{\theta}(\boldsymbol{x}, t) \int_0^{\tau_{\max}} d\tau + \sum_{k=1}^{2} \left(\int_0^{\tau_{\max}} D(\tau) B_k D(-\tau) d\tau \right) \partial_k \boldsymbol{\theta}(\boldsymbol{x}, t) \right) d\boldsymbol{x} dt = 0.$$

By integrating by parts, and because φ vanishes at $t=0$ and $t=T$, we have

$$\int_0^T \int_{\mathbb{R}^2} \left(\tau_{\max} \partial_t \boldsymbol{\mu}(\boldsymbol{x},t) + \sum_{k=1}^{2} \left(\int_0^{\tau_{\max}} D(\tau) B_k D(-\tau) d\tau \right) \partial_k \boldsymbol{\mu}(\boldsymbol{x},t) \right) \cdot \boldsymbol{\theta}(\boldsymbol{x},t) d\boldsymbol{x} dt = 0.$$

Thereby, $\boldsymbol{\mu}$ satisfies the equation

$$\tau_{\max} \partial_t \boldsymbol{\mu}(\boldsymbol{x},t) + \sum_{k=1}^{2} \left(\int_0^{\tau_{\max}} D(\tau) B_k D(-\tau) d\tau \right) \partial_k \boldsymbol{\mu}(\boldsymbol{x},t) = 0.$$

Finally, we have

$$\partial_t \boldsymbol{U}(\boldsymbol{x},t,\tau) + \sum_{k=1}^{2} M_k \partial_k \boldsymbol{U}(\boldsymbol{x},t,\tau) = 0,$$

where M_k are the matrices

$$M_k = \frac{1}{\tau_{\max}} D(\tau) \left(\int_0^{\tau_{\max}} D(\tau) B_k D(-\tau) d\tau \right) D(-\tau).$$

As $D(\tau)$ and B_k do not commute, we have

$$\begin{aligned} M_k &= \frac{1}{\tau_{\max}} D(\tau) \left(\int_0^{\tau_{\max}} \begin{pmatrix} B_{0,0}^k & e^{-\tau a} B_{0,1}^k & e^{-\tau a} B_{0,2}^k \\ e^{\tau a} B_{1,0}^k & B_{1,1}^k & B_{1,2}^k \\ e^{\tau a} B_{2,0}^k & B_{2,1}^k & B_{2,2}^k \end{pmatrix} d\tau \right) D(-\tau), \\ &= \begin{pmatrix} B_{0,0}^k & \gamma_2(\tau) B_{0,1}^k & \gamma_2(\tau) B_{0,2}^k \\ \gamma_1(\tau) B_{1,0}^k & B_{1,1}^k & B_{1,2}^k \\ \gamma_1(\tau) B_{2,0}^k & B_{2,1}^k & B_{2,2}^k \end{pmatrix}, \end{aligned}$$

with $\gamma_1(\tau) = \frac{e^{a\tau}}{\tau_{\max}} \left(\frac{e^{a\tau_{\max}}-1}{a} \right)$ and $\gamma_2(\tau) = -\frac{e^{-a\tau}}{\tau_{\max}} \left(\frac{e^{-a\tau_{\max}}-1}{a} \right)$. □

Remark 1.2. *As $a \leqslant 0$, we have $\lim_{\tau_{\max} \to +\infty} \gamma_1(\tau) = 0$, and $\lim_{\tau_{\max} \to +\infty} \gamma_2(\tau) = +\infty$. This is a little bit disappointing, because we would have preferred to obtain a simple limit system when $\tau_{\max}$ is large, which is obviously not the case here. This is probably because we have inserted in the analysis a too simple exponential behavior, given by formula (5.9). This behavior is a consequence of the choice of test functions satisfying (5.8). Obviously, this choice is not the best. The test function should be chosen in order to filter the fast behavior. It would probably be much more fruitful to choose test functions that isolate the behavior related to the rapidly decaying mode associated to eigenvalue $\alpha_1(k)$ (see Section (5.2)).*

Remark 1.3. *However, when $\omega = 2$, the two-scale analysis provides the good expected behavior. The terms of the non-diagonal blocs of the matrices B_k vanish.*

Consequently, in this case, we have

$$\partial_t \boldsymbol{U}(\boldsymbol{x},t,\tau) + \sum_{k=1}^{2} B_k \partial_k \boldsymbol{U}(\boldsymbol{x},t,\tau) = 0.$$

Moreover, the dumping term a cancels. Therefore, $D(\tau) = Id$ and the system does not depend on τ.

2 Conclusion

In this chapter, we have provided a tentative analysis of the equivalent system of the $D2Q3$ model with a two-scale convergence.

The two-scale limit gives the expected behavior in the case $\omega = 2$, with no dependency in the fast time variable τ.

In the case $\omega = 2$, our two-scale analysis does not give very useful information. We think that it is because our choice of test functions does not filter the good asymptotic behaviors when Δt is small. We think that it is possible to improve the analysis.

Chapter 6

Numerical stability

In chapter 4, we have computed the equivalent system on (w, Y) and the equivalent equation on w, for different kinetic models. In this chapter, we are interested in comparing the diffusive stability condition obtained from the equivalent equation and the hyperbolicity stability condition obtained from the equivalent system.

In the second part of this chapter, we will try to build stable boundary conditions of order 2 for these kinetic models.

1 Entropy stability of the kinetic model

1.1 Dual kinetic entropy representation

In this section, we recall the entropy theory of the kinetic representation. This theory has a long history, see for instance [63, 73, 53, 29, 16, 25, 77]. In our context it has been analyzed by Bouchut in [13]. However, in his work, Bouchut avoids the use of the Legendre transform. The ideas have been rephrased in an easier way (to our opinion) in the work of Dubois [35] with the help of the Legendre transform. Let us now recall the theory.

We consider a system of r conservation laws

$$\partial_t \boldsymbol{w} + \nabla \cdot \boldsymbol{q}(\boldsymbol{w}) = 0, \tag{6.1}$$

that admits a Lax entropy function $S(\boldsymbol{w})$ and a Lax entropy flux $\boldsymbol{G}(\boldsymbol{w})$ [63]. Thus $S : \mathbb{R}^r \to \mathbb{R}$ is strictly convex and

$$\partial_t S(\boldsymbol{w}) + \nabla \cdot \boldsymbol{G}(\boldsymbol{w}) = 0,$$

whenever $\boldsymbol{w}$ is a smooth solution of (6.1). This imposes that

$$D_{\boldsymbol{w}} S(\boldsymbol{w}) D_{\boldsymbol{w}} \boldsymbol{q}(\boldsymbol{w}) = D_{\boldsymbol{w}} \boldsymbol{G}(\boldsymbol{w}), \tag{6.2}$$

where we have denoted by $D_{\boldsymbol{w}} g(\boldsymbol{w})$ the Jacobian of $g(\boldsymbol{w})$. Let us recall that the Jacobian is the transpose of the gradient

$$D_{\boldsymbol{w}} S(\boldsymbol{w}) = \nabla_{\boldsymbol{w}} S(\boldsymbol{w})^\top. \tag{6.3}$$

Thus $D_{\boldsymbol{w}} S(\boldsymbol{w})$ is a row vector, while $\nabla_{\boldsymbol{w}} S(\boldsymbol{w})$ is a column vector.

As in Chapter 2, we consider a formal vectorial kinetic representation of that system

$$\partial_t \boldsymbol{f}_k + \boldsymbol{\lambda}_k \cdot \nabla \boldsymbol{f}_k = \frac{1}{\varepsilon}(\boldsymbol{f}_k^{eq} - \boldsymbol{f}_k), \quad k = 1, \ldots, n_v,$$

where the conservative vector is the sum of the kinetic vectors

$$\boldsymbol{w} = \sum_{k=1}^{n_v} \boldsymbol{f}_k,$$

and the kinetic equilibrium vectors (or Maxwellians) are functions of the conservative data

$$\boldsymbol{f}_k^{eq} = \boldsymbol{f}_k^{eq}(\boldsymbol{w}).$$

For the moment, instead of recovering the equilibrium from the conservation laws (6.1) as in Chapter 2, we assume that the equilibrium is obtained from an entropy optimization principle. For this we introduce a microscopic entropy

$$\Sigma(\boldsymbol{f}) = \sum_{k=1}^{n_v} s_k(\boldsymbol{f}_k),$$

where the kinetic entropies s_k are strictly convex functions of the $\boldsymbol{f}_k$. The macroscopic entropy is obtained from the resolution of the following constrained optimization problem

$$S(\boldsymbol{w}) = \min_{\boldsymbol{w}=\sum_k \boldsymbol{f}_k} \Sigma(\boldsymbol{f}). \tag{6.4}$$

In optimization this operation is known as an inf-convolution operation [59]. The macroscopic entropy is the inf-convolution of the kinetic entropies. In many works, the inf-convolution operator is denoted with a $\square$. We thus have:

$$S = s_1 \square s_2 \ldots \square s_{n_v}.$$

We denote by $\boldsymbol{f}_k^{eq}(\boldsymbol{w})$ the (supposed to be unique) values of $\boldsymbol{f}_k$ that achieve the minimum

$$S(\boldsymbol{w}) = \sum_{k=1}^{n_v} s_k(\boldsymbol{f}_k^{eq}(\boldsymbol{w})).$$

If we introduce the Lagrangian

$$L(\boldsymbol{f}, m) = \sum_{k=1}^{n_v} s_k(\boldsymbol{f}_k) + m \cdot (\boldsymbol{w} - \sum_{k=1}^{n_v} \boldsymbol{f}_k).$$

then the minimizer $\boldsymbol{f}^{eq}(\boldsymbol{w})$ and the Lagrange multiplier $m(\boldsymbol{w})$ are characterized by

$$\nabla_{\boldsymbol{f}_k} s_k(\boldsymbol{f}_k^{eq}) = m, \quad \sum_{k=1}^{n_v} \boldsymbol{f}_k^{eq} = \boldsymbol{w}. \tag{6.5}$$

These relations are simply obtained by deriving the Lagrangian with respect to $\boldsymbol{f}_k$ or m.

Let us introduce the Legendre transform [58, 59]

$$S^*(\boldsymbol{p}) = \max_{\boldsymbol{w}}(\boldsymbol{p}\cdot\boldsymbol{w} - S(\boldsymbol{w})). \tag{6.6}$$

The components of $\boldsymbol{p}$ are called the dual variables, or entropy variables [53, 16, 25]. The function S^* is called the dual entropy. The definition of the Legendre transform (6.6) applies to functions that are not necessarily smooth or convex. In the regular case, when S is smooth and strictly convex on $\mathbb{R}^r$, S^* is defined implicitly by the following relations

$$\boldsymbol{p} = \nabla S(\boldsymbol{w}(\boldsymbol{p})), \tag{6.7}$$

$$S^*(\boldsymbol{p}) = \boldsymbol{p}\cdot\boldsymbol{w}(\boldsymbol{p}) - S(\boldsymbol{w}(\boldsymbol{p})),$$

where we denote by $\cdot$ the usual dot product:

$$\boldsymbol{p}\cdot\boldsymbol{w}(\boldsymbol{p}) = \boldsymbol{p}^\top\boldsymbol{w}(\boldsymbol{p}) = \boldsymbol{w}(\boldsymbol{p})^\top\boldsymbol{p}.$$

In this case, it can also be shown that $S^{**} = S$. Thus we have the reverse relations:

$$\boldsymbol{w} = \nabla S^*(\boldsymbol{p}(\boldsymbol{w})),$$

$$S(\boldsymbol{w}) = \boldsymbol{p}(\boldsymbol{w})\cdot\boldsymbol{w} - S^*(\boldsymbol{p}(\boldsymbol{w})).$$

We can also define the dual entropy flux by the relation

$$G^{i,\star}(\boldsymbol{p}) = \boldsymbol{p}\cdot q^i(\boldsymbol{w}(\boldsymbol{p})) - G^i(\boldsymbol{w}(\boldsymbol{p})). \tag{6.8}$$

Let us remark that we do not use the same symbol for the dual entropy ($*$) and the dual flux ($\star$), because the definitions are different. An important fact is that the knowledge of $S^*(\boldsymbol{p})$ and $G^{i,\star}(\boldsymbol{p})$ is sufficient to reconstruct the system of conservation laws (6.1). Indeed,

$$\begin{aligned}\nabla_{\boldsymbol{p}}G^{i,\star}(\boldsymbol{p}) &= q^i(\boldsymbol{w}(\boldsymbol{p})) + \boldsymbol{p}\cdot D_{\boldsymbol{w}}q^i(\boldsymbol{w}(\boldsymbol{p}))D_{\boldsymbol{p}}\boldsymbol{w}(\boldsymbol{p}) - D_{\boldsymbol{w}}G^i(\boldsymbol{w}(\boldsymbol{p}))D_{\boldsymbol{p}}\boldsymbol{w}(\boldsymbol{p}),\\ &= q^i(\boldsymbol{w}(\boldsymbol{p})) + D_{\boldsymbol{w}}S(\boldsymbol{w})D_{\boldsymbol{w}}q^i(\boldsymbol{w}(\boldsymbol{p}))D_{\boldsymbol{p}}\boldsymbol{w}(\boldsymbol{p}) - D_{\boldsymbol{w}}G^i(\boldsymbol{w}(\boldsymbol{p}))D_{\boldsymbol{p}}\boldsymbol{w}(\boldsymbol{p}),\end{aligned}$$

(because of (6.7) and (6.3))

$$= q^i(\boldsymbol{w}(\boldsymbol{p})) + D_{\boldsymbol{w}}G^i(\boldsymbol{w}(\boldsymbol{p}))D_{\boldsymbol{p}}\boldsymbol{w}(\boldsymbol{p}) - D_{\boldsymbol{w}}G^i(\boldsymbol{w}(\boldsymbol{p}))D_{\boldsymbol{p}}\boldsymbol{w}(\boldsymbol{p}),$$

(because of (6.2)), and thus

$$\nabla_{\boldsymbol{p}}G^{i,\star}(\boldsymbol{p}) = q^i(\boldsymbol{w}(\boldsymbol{p})). \tag{6.9}$$

In short: the gradient of the dual entropy gives the conservative variables and the gradient of the dual flux gives the flux.

Remark: an important remark is that the change of variables $\boldsymbol{p} \mapsto \boldsymbol{w}(\boldsymbol{p})$ is a way to symmetrize the system of conservation laws. Indeed

$$\partial_t\boldsymbol{w} + \partial_x\boldsymbol{q}(\boldsymbol{w}) = 0,$$

can be rewritten

$$\partial_t\nabla_{\boldsymbol{p}}S^*(\boldsymbol{p}) + \sum_i \partial_i\nabla_{\boldsymbol{p}}G^{i,\star}(\boldsymbol{p}) = 0,$$

or

$$D_{\boldsymbol{pp}}S^*(\boldsymbol{p})\partial_t\boldsymbol{p} + \sum_i D_{\boldsymbol{pp}}G^{i,\star}(\boldsymbol{p})\partial_i\boldsymbol{p} = 0, \tag{6.10}$$

where the Hessian matrices $D_{\boldsymbol{pp}}S^*(\boldsymbol{p})$ and $D_{\boldsymbol{pp}}G^{i,\star}(\boldsymbol{p})$ are obviously symmetric and $D_{\boldsymbol{pp}}S^*(\boldsymbol{p})$ is positive definite (when S^* is strictly convex). This is the Mock theorem [73]. In the following sections, we shall often try to guess directly a symmetrization of the system of conservation laws, rather than the full dual entropy theory to obtain the stability conditions of the relaxation scheme.

An essential property of the inf-convolution is that the Legendre transform changes it into a sum. We thus have

$$S^*(\boldsymbol{p}) = \sum_{k=1}^{n_v} s_k^*(\boldsymbol{p}).$$

Taking the Legendre transform of (6.5) we see that the couple $(\boldsymbol{f}_k^{eq}(\boldsymbol{w}), m(\boldsymbol{w}))$ is solution to

$$\boldsymbol{f}_k^{eq} = \nabla_m s_k^*(m(\boldsymbol{w})), \quad \sum_{k=1}^{n_v} \boldsymbol{f}_k^{eq} = \boldsymbol{w}. \tag{6.11}$$

Summing over k, we also have the Lagrange multiplier in an easier way

$$\sum_{k=1}^{n_v} \boldsymbol{f}_k^{eq} = \sum_{k=1}^{n_v} \nabla_m s_k^*(m(\boldsymbol{w})),$$

which gives us

$$\begin{aligned}\boldsymbol{w} &= \nabla_m \sum_{k=1}^{n_v} s_k^*(m(\boldsymbol{w})),\\ &= \nabla_m S^*(m(\boldsymbol{w})),\end{aligned}$$

and thus

$$m(\boldsymbol{w}) = \boldsymbol{p}(\boldsymbol{w}) = \nabla S(\boldsymbol{w}).$$

It is remarkable that the Lagrange multiplier of the constrained optimization problem (6.5) is simply the gradient of the macroscopic entropy.

Let us now assume the additional property

$$G^{i,\star}(\boldsymbol{p}) = \sum_{k=1}^{n_v} \lambda_k^i s_k^*(\boldsymbol{p}). \tag{6.12}$$

Then we have

$$\begin{aligned}\nabla_{\boldsymbol{p}} G^{i,\star}(\boldsymbol{p}) &= \sum_{k=1}^{n_v} \lambda_k^i \nabla_{\boldsymbol{p}} s_k^*(\boldsymbol{p}),\\ \nabla_{\boldsymbol{p}} G^{i,\star}(\boldsymbol{p}) &= \sum_{k=1}^{n_v} \lambda_k^i \boldsymbol{f}_k^{eq}(\boldsymbol{w}),\end{aligned}$$

(from (6.11)) and thus, from (6.9)

$$q^i(\boldsymbol{w}) = \sum_{k=1}^{n_v} \lambda_k^i \boldsymbol{f}_k^{eq}(\boldsymbol{w}). \tag{6.13}$$

We recover relation (2.6) that imposes the consistency of the kinetic model with the system of conservation laws.

1.2 Reverse construction

Now that we have recalled the entropy theory of the kinetic representation, we can proceed in the reverse way. We *choose* the equilibrium $\boldsymbol{f}^{eq}$ in such a way that the consistency relation (6.13) is satisfied. In the previous chapters, we have seen that if we take only $d+1$ kinetic velocities $\boldsymbol{\lambda}_k$ then this choice is generally unique. This is the case of the $D1Q2$ and $D2Q3$ models. From the above theory, we know that $\boldsymbol{f}_k^{eq}(\boldsymbol{w}(\boldsymbol{p}))$ is a gradient, when it is expressed in the entropy variables $\boldsymbol{p}$! We can thus find dual kinetic entropies $s_k^*(\boldsymbol{p})$ such that

$$\boldsymbol{f}_k^{eq}(\boldsymbol{w}(\boldsymbol{p})) = \nabla_{\boldsymbol{p}} s_k^*(\boldsymbol{p}).$$

By Legendre transform, we can (in principle) compute the kinetic entropies $s_k(\boldsymbol{f}_k)$ and this gives us the microscopic entropy

$$\Sigma(\boldsymbol{f}) = \sum_{k=1}^{n_v} s_k(\boldsymbol{f}_k).$$

The main point in the reverse construction is to ensure that the strict convexity is preserved. In practice, we will see that the microscopic entropy is convex under a subcharacteristic condition.

1.3 Application to the transport equation

Let us now try to apply the above construction to the $D1Q2$ model for the transport equation. In this case

$$\boldsymbol{q}(\boldsymbol{w}) = v\boldsymbol{w}, \quad \lambda_1 = \lambda, \quad \lambda_2 = -\lambda,$$

and we have no free choice for choosing the equilibrium kinetic data, which are given by

$$\boldsymbol{f}_1^{eq}(\boldsymbol{w}) = \frac{\boldsymbol{w}}{2} + \frac{v\boldsymbol{w}}{2\lambda}, \quad \boldsymbol{f}_2^{eq}(\boldsymbol{w}) = \frac{\boldsymbol{w}}{2} - \frac{v\boldsymbol{w}}{2\lambda}.$$

For this simple linear conservation law we can take the entropy associated to the L^2 norm

$$S(\boldsymbol{w}) = \frac{\boldsymbol{w}^2}{2}.$$

The dual entropy is simply

$$S^*(\boldsymbol{p}) = \frac{\boldsymbol{p}^2}{2}.$$

and the entropy variable is

$$\boldsymbol{p} = \nabla_{\boldsymbol{w}} S(\boldsymbol{w}) = \boldsymbol{w}.$$

Thus

$$\boldsymbol{f}_1^{eq}(\boldsymbol{w}(\boldsymbol{p})) = \frac{\boldsymbol{p}}{2} + \frac{v\boldsymbol{p}}{2\lambda}, \quad \boldsymbol{f}_2^{eq}(\boldsymbol{w}(\boldsymbol{p})) = \frac{\boldsymbol{p}}{2} - \frac{v\boldsymbol{p}}{2\lambda}.$$

From (6.11) we deduce the dual kinetic entropies

$$s_1^*(\boldsymbol{p}) = \frac{1}{4}(1 + v/\lambda)\boldsymbol{p}^2, \quad s_2^*(\boldsymbol{p}) = \frac{1}{4}(1 - v/\lambda)\boldsymbol{p}^2.$$

They are strictly convex under the subcharacteristic condition

$$\lambda > |v|\,. \tag{6.14}$$

We can then compute the kinetic entropies

$$s_1(\boldsymbol{f}_1) = \frac{\lambda}{\lambda + v}\boldsymbol{f}_1^2, \quad s_2(\boldsymbol{f}_2) = \frac{\lambda}{\lambda - v}\boldsymbol{f}_2^2.$$

The microscopic entropy is then

$$\Sigma(\boldsymbol{f}_1, \boldsymbol{f}_2) = \frac{\lambda}{\lambda + v}\boldsymbol{f}_1^2 + \frac{\lambda}{\lambda - v}\boldsymbol{f}_2^2.$$

As expected, it is a diagonal quadratic form in the $(\boldsymbol{f}_1, \boldsymbol{f}_2)$ variables.

Let us express the microscopic entropy with respect to the $(\boldsymbol{w}, \boldsymbol{y})$ variables. We have

$$\boldsymbol{w} = \boldsymbol{f}_1 + \boldsymbol{f}_2,$$

and

$$\begin{aligned} \boldsymbol{y} &= \lambda\boldsymbol{f}_1 - \lambda\boldsymbol{f}_2 - \boldsymbol{q}(\boldsymbol{w}), \\ &= \lambda\boldsymbol{f}_1 - \lambda\boldsymbol{f}_2 - v(\boldsymbol{f}_1 + \boldsymbol{f}_2). \end{aligned}$$

After simple computations, we find that the microscopic entropy is also

$$\widetilde{\Sigma}(\boldsymbol{w}, \boldsymbol{y}) = \Sigma(\boldsymbol{f}_1, \boldsymbol{f}_2) = \frac{\boldsymbol{w}^2}{2} + \frac{\boldsymbol{y}^2}{2(\lambda^2 - v^2)}. \tag{6.15}$$

It is a convex function of $\boldsymbol{w}$ and $\boldsymbol{y}$ under the subcharacteristic condition (6.14). As expected, it is minimal when the flux error $\boldsymbol{y}$ vanishes. In addition, in the relaxation step, the entropy is exactly conserved when $\omega = 2$ because

$$\widetilde{\Sigma}(\boldsymbol{w}, (1 - \omega)\boldsymbol{y}) = \widetilde{\Sigma}(\boldsymbol{w}, -\boldsymbol{y}) = \widetilde{\Sigma}(\boldsymbol{w}, \boldsymbol{y}). \tag{6.16}$$

We are now in a position to prove the entropy stability of the over-relaxation scheme when $1 \leqslant \omega \leqslant 2$.

Theorem 1.1. *With periodic boundary conditions, or in an infinite domain, the over-relaxation scheme is entropy stable under the sub-characteristic condition* (6.14) *when* $1 \leqslant \omega \leqslant 2$.

Proof. It is sufficient to prove the decrease of the entropy

$$\mathscr{S}(t) = \int_x \Sigma(\boldsymbol{f}_1(x,t), \boldsymbol{f}_2(x,t)) = \int_x s_1(\boldsymbol{f}_1(x,t)) + s_2(\boldsymbol{f}_2(x,t)),$$

for a single time step. In the transport step, one solves

$$\partial_t \boldsymbol{f}_k + \lambda_k \partial_x \boldsymbol{f}_k = 0,$$

and thus the microscopic entropies

$$\mathscr{S}_k(t) = \int_x s_k(\boldsymbol{f}_k(x,t)),$$

are separately conserved

$$\mathscr{S}_k(t + \Delta t^-) = \mathscr{S}_k(t^+).$$

In the relaxation step, $\boldsymbol{w}$ is not changed and

$$\boldsymbol{y}(x, t + \Delta t^+) = (1 - \omega)\boldsymbol{y}(x, t + \Delta t^-),$$

because $|1 - \omega| \leqslant 1$ we see from the expression (6.15) of the entropy in the $(\boldsymbol{w}, \boldsymbol{y})$ variables that the microscopic entropy decreases pointwise, at each x. Therefore

$$\mathscr{S}(t + \Delta t^+) \leqslant \mathscr{S}(t + \Delta t^-).$$

□

1.4 Application to a truly nonlinear system

In order to show that the approach still works for a non-linear system of conservation laws, we try now to apply the above method to the $D1Q2$ model for the shallow water model where the unknowns are the water height $h(x,t)$ and the velocity $u(x,t)$. It reads

$$\partial_t \boldsymbol{w} + \partial_x \boldsymbol{q}(\boldsymbol{w}) = 0,$$

with

$$\boldsymbol{w} = \begin{pmatrix} h \\ hu \end{pmatrix}, \quad \boldsymbol{q}(\boldsymbol{w}) = \begin{pmatrix} hu \\ hu^2 + gh^2/2 \end{pmatrix}, \quad g = 9.81\mathrm{m/s}^2.$$

We define the primitive variables

$$\boldsymbol{v} = \begin{pmatrix} h \\ u \end{pmatrix}.$$

For smooth solutions, we also have

$$\partial_t \boldsymbol{v} + B(\boldsymbol{v})\partial_x \boldsymbol{v} = 0,$$

with

$$B(\boldsymbol{v}) = \begin{pmatrix} u & h \\ g & u \end{pmatrix}.$$

Assume that the Lax entropy $S(\boldsymbol{w}) = H(\boldsymbol{v})$ is expressed in the primitive variables, and that the entropy flux $G(\boldsymbol{w}) = R(\boldsymbol{v})$. Then we must have

$$D_{\boldsymbol{v}} H(\boldsymbol{v}) B(\boldsymbol{v}) = D_{\boldsymbol{v}} R(\boldsymbol{v}).$$

Denoting the partial derivatives with indices we obtain

$$\begin{pmatrix} H_h & H_u \end{pmatrix} \begin{pmatrix} u & h \\ g & u \end{pmatrix} = \begin{pmatrix} R_h & R_u \end{pmatrix}.$$

We search H under the form

$$H(h,u) = h\frac{u^2}{2} + e(h).$$

Because

$$H_h = \frac{u^2}{2} + e'(h), \quad H_u = hu,$$

this gives

$$\frac{u^3}{2} + ue' + ghu = R_h, \quad \frac{3hu^2}{2} + he' = R_u.$$

We take

$$R = h\frac{u^3}{2} + ue + gu\frac{h^2}{2}.$$

Then

$$R_u = \frac{3hu^2}{2} + e + g\frac{h^2}{2} = \frac{3hu^2}{2} + he'.$$

$e(h)$ is then solution of the differential equation

$$e - he' + gh^2/2 = 0.$$

A solution is

$$e(h) = \frac{gh^2}{2}.$$

In the end we find

$$S(\boldsymbol{w}) = h\frac{u^2}{2} + \frac{gh^2}{2}, \quad G(\boldsymbol{w}) = h\frac{u^3}{2} + ugh^2.$$

This allows us to compute the entropy variables

$$p_1 = gh - \frac{u^2}{2}, \quad p_2 = u, \tag{6.17}$$

and the reverse change of variables

$$h = \frac{2p_1 + p_2^2}{2g}, \quad u = p_2.$$

The equilibrium kinetic vectors are given by

$$\boldsymbol{f}_1^{eq} = \frac{\boldsymbol{w}}{2} + \frac{\boldsymbol{q}(\boldsymbol{w})}{2\lambda}, \quad \boldsymbol{f}_2^{eq} = \frac{\boldsymbol{w}}{2} - \frac{\boldsymbol{q}(\boldsymbol{w})}{2\lambda}.$$

After some calculations, we can express this equilibrium in the entropy variables

$$\boldsymbol{f}_1^{eq} = \begin{bmatrix} \frac{\left({p_2}^2+2p_1\right)(\lambda+p_2)}{4g\lambda} & \frac{\left({p_2}^2+2p_1\right)\left(4p_2\lambda+5{p_2}^2+2p_1\right)}{16g\lambda} \end{bmatrix}^\top,$$

$$\boldsymbol{f}_2^{eq} = \begin{bmatrix} \frac{\left({p_2}^2+2p_1\right)(\lambda-p_2)}{4g\lambda} & -\frac{\left({p_2}^2+2p_1\right)\left(-4p_2\lambda+5{p_2}^2+2p_1\right)}{16g\lambda} \end{bmatrix}^\top.$$

From the above theory, we know that

$$\boldsymbol{f}_k^{eq} = \nabla_{\boldsymbol{p}} s_k^*$$

for some dual kinetic entropies s_k^*. This is indeed the case and after more calculations we find

$$s_1^* = \frac{\left(p_2{}^2 + 2p_1\right)^2 (\lambda + p_2)}{16g\lambda}, \quad s_2^* = \frac{(\lambda - p_2)\left(p_2{}^2 + 2p_1\right)^2}{16g\lambda}.$$

It is then possible to compute the Hessians of s_k^* and express them in the (h, u) variables with (6.17). We find

$$D_{\boldsymbol{pp}} s_1^* = \begin{bmatrix} \frac{\lambda+u}{2g\lambda} & \frac{gh+\lambda u+u^2}{2g\lambda} \\ \frac{gh+\lambda u+u^2}{2g\lambda} & \frac{\left(gh+u^2\right)\lambda+3hug+u^3}{2g\lambda} \end{bmatrix},$$

$$D_{\boldsymbol{pp}} s_2^* = \begin{bmatrix} \frac{\lambda-u}{2g\lambda} & \frac{-gh+\lambda u-u^2}{2g\lambda} \\ \frac{-gh+\lambda u-u^2}{2g\lambda} & \frac{\left(gh+u^2\right)\lambda-3hug-u^3}{2g\lambda} \end{bmatrix}.$$

The two matrices are positive definite if and only if the first diagonal terms and the determinants are positive. This is equivalent to

$$\lambda > |u|,$$

$$(\lambda - u)^2 - gh > 0, \quad (\lambda + u)^2 - gh > 0,$$

which is again equivalent to

$$\lambda > |u| + \sqrt{gh}.$$

This is the expected sub-characteristic condition. It is difficult to go further because the Legendre transforms s_1 and s_2 of s_1^* and s_2^* are difficult to compute explicitly. However, we can reproduce the stability proof of the linear case. The microscopic entropy is given by

$$\Sigma(\boldsymbol{f}_1, \boldsymbol{f}_2) = s_1(\boldsymbol{f}_1) + s_2(\boldsymbol{f}_2).$$

Using the relations

$$\boldsymbol{w} = \boldsymbol{f}_1 + \boldsymbol{f}_2,$$

$$\boldsymbol{y} = \lambda \boldsymbol{f}_1 - \lambda \boldsymbol{f}_2 - \boldsymbol{q}(\boldsymbol{f}_1 + \boldsymbol{f}_2),$$

we deduce

$$\boldsymbol{f}_1 = \frac{\boldsymbol{w}}{2} + \frac{\boldsymbol{q}(\boldsymbol{w})}{2\lambda} + \frac{\boldsymbol{y}}{2\lambda}, \quad \boldsymbol{f}_2 = \frac{\boldsymbol{w}}{2} - \frac{\boldsymbol{q}(\boldsymbol{w})}{2\lambda} - \frac{\boldsymbol{y}}{2\lambda},$$

and the microscopic entropy can be expressed in function of $\boldsymbol{w}$ and $\boldsymbol{y}$

$$\widetilde{\Sigma}(\boldsymbol{w}, \boldsymbol{y}) = s_1\left(\frac{\boldsymbol{w}}{2} + \frac{\boldsymbol{q}(\boldsymbol{w})}{2\lambda} + \frac{\boldsymbol{y}}{2\lambda}\right) + s_2\left(\frac{\boldsymbol{w}}{2} - \frac{\boldsymbol{q}(\boldsymbol{w})}{2\lambda} - \frac{\boldsymbol{y}}{2\lambda}\right).$$

For a fixed $\boldsymbol{w}$, the minimum of the entropy is achieved for $\boldsymbol{y} = 0$ since it corresponds to the equilibria distribution associated to $\boldsymbol{w}$. Therefore the macroscopic entropy is

$$S(\boldsymbol{w}) = \widetilde{\Sigma}(\boldsymbol{w}, 0).$$

and

$$\nabla_{\boldsymbol{y}}\widetilde{\Sigma}(\boldsymbol{w},0)=0.$$

Then, with a Taylor expansion near to $\boldsymbol{y}=0$, we get

$$\widetilde{\Sigma}(\boldsymbol{w},\boldsymbol{y})=\widetilde{\Sigma}(\boldsymbol{w},-\boldsymbol{y})+O(|\boldsymbol{y}|^3).$$

The relation (6.16) thus still holds but with a third-order term in $\boldsymbol{y}$. This means that the relaxation scheme with $\omega=2$ is entropy preserving up to third order in $\boldsymbol{y}$. In principle, it is also possible to construct a scheme that preserves exactly the entropy in the non-linear case. It is sufficient to choose the relaxation parameter $\omega=\omega(\boldsymbol{w},\boldsymbol{y})$ in such way that

$$\widetilde{\Sigma}(\boldsymbol{w},(1-\omega(\boldsymbol{w},\boldsymbol{y}))\boldsymbol{y})=\widetilde{\Sigma}(\boldsymbol{w},\boldsymbol{y}). \tag{6.18}$$

In practice, this would not be very interesting, one would get

$$\omega(\boldsymbol{w},\boldsymbol{y})\simeq 2,$$

and $\omega(\boldsymbol{w},\boldsymbol{y})$ would have to be computed numerically by first computing s_1 and s_2 numerically and then by solving (6.18) also numerically.

What is interesting, however, is that the reasoning ensures the existence of a relaxation parameter $\omega(\boldsymbol{w},\boldsymbol{y})\simeq 2$, such that the whole scheme is entropy preserving. And if the scheme is run with a smaller relaxation parameter, it is ensured to be entropy stable.

2 Stability conditions

We now consider $w\in\mathbb{R}$.

To analyze the stability of our kinetic model, we propose to compare two stability conditions. First, we recall the classical diffusive stability condition, obtained by satisfying the positivity of the diffusion term of the equivalent equation on w.

Then, we propose to compute the condition for our equivalent system on $(w,\boldsymbol{Y})$ to be hyperbolic. More precisely, we are searching for a matrix that symmetries the equivalent system. If such a matrix exists, the system is hyperbolic. We remind the proof of this theorem in the following subsection.

2.1 Hyperbolicity condition

We consider a system of the form

$$\partial_t v+\sum_{i=1}^{d}B_i\partial_i v=0. \tag{6.19}$$

Definition 2.1. *The system (6.19) is hyperbolic if for all unit vector $\boldsymbol{n}\in\mathbb{R}^d$, the matrix $\sum_{i=1}^d n_iB_i$ is diagonalizable in $\mathbb{R}$.*

Definition 2.2. *The system (6.19) is symmetrizable if it exists a symmetric positive definite matrix P such as the matrix $P\left(\sum_{i=1}^{d} n_i B_i\right)$ are symmetric.*

Remark 2.1. *We can simplify the previous definition: The system (6.19) is symmetrizable if it exists a symmetric positive definite matrix P such as the matrices PB_i are symmetric for all $i = 1, \ldots, d$.*

Theorem 2.1. *A symetrizable system is hyperbolic.*

Proof. Let us assume that the system (6.19) is symmetrizable. We denote $B(\boldsymbol{n}) = \sum_{i=1}^{d} n_i B_i \in M_d(\mathbb{R})$. Let us note R the symmetric positive definite matrix such as $R^2 = P^{-1}$. We have

$$B(\boldsymbol{n}) = P^{-1}PB(\boldsymbol{n}) = R^2 PB(\boldsymbol{n}) = R(RPB(\boldsymbol{n})R)R^{-1}.$$

As $PB(\boldsymbol{n})$ and R are symmetric, $RPB(\boldsymbol{n})R$ is symmetric. Then $RPB(\boldsymbol{n})R$ is diagonalizable in $\mathbb{R}$. It exists an orthogonal $Q(\boldsymbol{n})$ and a real diagonal matrix $D(\boldsymbol{n})$ such as

$$RPB(\boldsymbol{n})R = Q(\boldsymbol{n})D(\boldsymbol{n})Q(\boldsymbol{n})^\top.$$

Consequently, we have

$$\begin{aligned} B(\boldsymbol{n}) &= R(Q(\boldsymbol{n})D(\boldsymbol{n})Q(\boldsymbol{n})^\top)R^{-1}, \\ &= (RQ(\boldsymbol{n}))D(\boldsymbol{n})(RQ)^{-1}. \end{aligned}$$

Finally, $B(\boldsymbol{n})$ is diagonalizable in $\mathbb{R}$. The system (6.19) is then hyperbolic. □

2.2 The $D1Q2$ model

Diffusive stability condition

We consider a linear flux $q(w) = vw$.

Proposition 2.1. *When $\omega \neq 2$, the sub-characteristic diffusive stability condition of the $D1Q2$ model is*

$$|v| < \lambda.$$

Proof. In chapter 4, we have computed the equivalent equation on w for the $D1Q2$ model and obtained the equation (4.8)

$$\partial_t w + v\partial_x w = \frac{1}{2}\left(\frac{1}{\omega} - \frac{1}{2}\right)(\lambda^2 - v^2)\Delta t \partial_{xx} w + O(\Delta t^2).$$

The equation is stable if the diffusion term is positive. As $\omega \in [1, 2]$, the term $\left(\frac{1}{\omega} - \frac{1}{2}\right)$ is positive. The positivity of the diffusion term is then equivalent to

$$\lambda^2 - v^2 > 0,$$

which gives us the stability condition

$$|v| < \lambda.$$

□

Remark 2.2. *When $\omega = 2$, the diffusion term of the equivalent equation of the $D1Q2$ model disappears, which gives us*

$$\partial_t w + v\partial_x w = O(\Delta t^2).$$

We obtain that the solution given by the $D1Q2$ model is an approximation of order 2 of the solution of the initial equation.

Remark 2.3. *We retrieve the subcharacteristic stability condition (6.14) given by the entropy stability analysis.*

Hyperbolicity condition

In Chapter 4, we have found equivalent system of the form

$$\partial_t \begin{pmatrix} w \\ y \end{pmatrix} - \frac{a}{\Delta t}\begin{pmatrix} 0 \\ y \end{pmatrix} + B\partial_x \begin{pmatrix} w \\ y \end{pmatrix} = O(\Delta t). \tag{6.20}$$

We can also write it

$$\partial_t \begin{pmatrix} w \\ y \end{pmatrix} - \frac{A}{\Delta t}\begin{pmatrix} 0 \\ y \end{pmatrix} + B\partial_x \begin{pmatrix} w \\ y \end{pmatrix} = O(\Delta t), \tag{6.21}$$

with $A = \begin{pmatrix} 0 & 0 \\ 0 & a \end{pmatrix}$.

Let us note

$$v = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{at}{\Delta t}} \end{pmatrix}\begin{pmatrix} w \\ y \end{pmatrix}.$$

We have

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{at}{\Delta t}} \end{pmatrix}\left(\partial_t \begin{pmatrix} w \\ y \end{pmatrix} - \frac{A}{\Delta t}\begin{pmatrix} 0 \\ y \end{pmatrix} + B\partial_x \begin{pmatrix} w \\ y \end{pmatrix}\right) = 0, \tag{6.22}$$

which gives us

$$\partial_t v + B\partial_x v = 0.$$

Now that we do not have any damping term, we can study the hyperbolicity of this system.

Proposition 2.2. *The matrix*

$$P = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\lambda^2 - v^2} \end{pmatrix},$$

symmetrizes the equivalent system of the $D1Q2$ model (4.6), if the diffusive sub-characteristic stability condition is satisfied. Consequently, the equivalent system (4.6) is hyperbolic if

$$|v| < \lambda.$$

Proof. We search a matrix $P = \begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix}$ such as PB is symmetric and P is symmetric positive definite. We have

$$\begin{aligned} PB &= \begin{pmatrix} p_1 & p_2 \\ p_2 & p_3 \end{pmatrix}\begin{pmatrix} v & \gamma_1 \\ (\lambda^2 - v^2)\gamma_1 & -v\gamma_2 \end{pmatrix}, \\ &= \begin{pmatrix} vp_1 + (\lambda^2 - v^2)\gamma_1 p_2 & \gamma_1 p_1 - v\gamma_2 p_2 \\ vp_2 + (\lambda^2 - v^2)\gamma_1 p_3 & \gamma_1 p_2 - v\gamma_2 p_3 \end{pmatrix}. \end{aligned}$$

with $\gamma_1 = \frac{(\omega-2)^2(\omega^2-2\omega+2)}{8(\omega-1)^2}$, and $\gamma_2 = \frac{(\omega^4-4\omega^3+6\omega^2-4\omega+2)}{2(\omega-1)^2}$.

As we want PB to be symmetric, we need to satisfy the condition

$$\gamma_1 p_1 - v\gamma_2 p_2 = vp_2 + (\lambda^2 - v^2)\gamma_1 p_3,$$

which is equivalent to

$$p_3 = \frac{1}{(\lambda^2 - v^2)} p_1 - v\frac{1+\gamma_2}{(\lambda^2 - v^2)\gamma_1} p_2.$$

Let us choose $p_2 = 0$ and $p_1 = 1$. We obtain

$$p_3 = \frac{1}{\lambda^2 - v^2},$$

and then

$$P = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\lambda^2 - v^2} \end{pmatrix}.$$

As its eigenvalues are 1 and $\frac{1}{\lambda^2-v^2}$, P is definite positive if

$$|v| < \lambda.$$

□

Remark 2.4. *We obtain the same condition on v and λ as for the diffusive stability condition given in Proposition 2.1. In this case, the diffusive analysis and the hyperbolicity analysis give the same stability condition.*

2.3 The $D2Q3$ model

Diffusive stability condition

We consider a linear flux $\boldsymbol{q}(w) = \begin{pmatrix} v_1 w \\ v_2 w \end{pmatrix}$.

Proposition 2.3. *The sub-characteristic stability condition of the $D2Q3$ model is*

$$\lambda^2 - {v_1}^2 - {v_2}^2 - \sqrt{({v_1}^2 + {v_2}^2)^2 + \lambda(-2{v_1}^3 + 6v_1{v_2}^2) + \lambda^2({v_1}^2 + {v_2}^2)} > 0.$$

Remark 2.5. *This condition can also be written as the intersection of*

$${v_1}^2 + {v_2}^2 < \lambda^2,$$

and

$$\lambda^3 - 3\lambda({v_1}^2 + {v_2}^2) + 2{v_1}^3 - 6v_1{v_2}^2 > 0.$$

Proof. Indeed, with a linear flux, we have

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = \frac{\Delta t}{2}\left(\frac{1}{\omega} - \frac{1}{2}\right)\nabla \cdot (\mathcal{D}_3 \nabla w) + O(\Delta t^2),$$

with the diffusion matrix

$$\mathcal{D}_3 = \begin{pmatrix} \frac{\lambda}{2}(\lambda + v_1) - {v_1}^2 & -\frac{\lambda}{2}v_2 - v_1 v_2 \\ -\frac{\lambda}{2}v_2 - v_1 v_2 & \frac{\lambda}{2}(\lambda - v_1) - {v_2}^2 \end{pmatrix}.$$

The eigenvalues of this diffusion matrix are

$$d_{1,2} = \frac{1}{2}\left(\lambda^2 - {v_1}^2 - {v_2}^2 \pm \sqrt{({v_1}^2 + {v_2}^2)^2 + \lambda(-2{v_1}^3 + 6v_1{v_2}^2) + \lambda^2({v_1}^2 + {v_2}^2)}\right).$$

Finally, the model $D2Q3$ is stable if $\mathcal{D}_3$ is positive definite, namely if $d_1 > 0$ and $d_2 > 0$. □

Hyperbolicity condition

Proposition 2.4. *The matrix*

$$P = \begin{pmatrix} \frac{\lambda}{2}({v_1}^2 - 2v_1\lambda - 3{v_2}^2 + \lambda^2)(2v_1 + \lambda) & 0 & 0 \\ 0 & -(v_1\lambda + 2{v_2}^2 - \lambda^2) & v_2(2v_1 + \lambda) \\ 0 & v_2(2v_1 + \lambda) & -(v_1 - \lambda)(2v_1 + \lambda) \end{pmatrix}.$$

symmetrizes the equivalent system of the $D2Q3$ model (4.15), if the diffusive sub-characteristic stability condition (2.3) is verified. Consequently, the equivalent system (4.15) is hyperbolic if

$$\lambda^2 - {v_1}^2 - {v_2}^2 - \sqrt{({v_1}^2 + {v_2}^2)^2 + \lambda(-2{v_1}^3 + 6v_1{v_2}^2) + \lambda^2({v_1}^2 + {v_2}^2)} > 0.$$

Proof.

We are searching for a matrix $P = \begin{pmatrix} p_1 & p_2 & p_3 \\ p_2 & p_4 & p_5 \\ p_3 & p_5 & p_6 \end{pmatrix}$ such as PB_1 and PB_2 are symmetric and P is symmetric positive definite. When we compute the matrices PB_1 and PB_2, the symmetry imposes 6 equations on the unknown $p_1, p_2, p_3, p_4, p_5, p_6$. This gives us the matrix

$$P = \begin{pmatrix} \frac{\lambda}{2}\frac{({v_1}^2 - 2v_1\lambda - 3{v_2}^2 + \lambda^2)p_5}{v_2} & 0 & 0 \\ 0 & -\frac{(v_1\lambda + 2{v_2}^2 - \lambda^2)p_5}{v_2(2v_1 + \lambda)} & p_5 \\ 0 & p_5 & -\frac{(v_1 - \lambda)p_5}{v_2} \end{pmatrix},$$

where p_5 must be chosen. We choose $p_5 = v_2(2v_1 + \lambda)$. We obtain

$$P = \begin{pmatrix} \frac{\lambda}{2}({v_1}^2 - 2v_1\lambda - 3{v_2}^2 + \lambda^2)(2v_1 + \lambda) & 0 & 0 \\ 0 & -(v_1\lambda + 2{v_2}^2 - \lambda^2) & v_2(2v_1 + \lambda) \\ 0 & v_2(2v_1 + \lambda) & -(v_1 - \lambda)(2v_1 + \lambda) \end{pmatrix}.$$

The eigenvalues of P are

$$e_1 = \frac{\lambda}{2}({v_1}^2 - 2v_1\lambda - 3{v_2}^2 + \lambda^2)(2v_1 + \lambda),$$

$$e_2 = \lambda^2 - {v_1}^2 - {v_2}^2 + \sqrt{({v_1}^2 + {v_2}^2)^2 + \lambda(-2{v_1}^3 + 6v_1{v_2}^2) + \lambda^2({v_1}^2 + {v_2}^2)},$$

and

$$e_3 = \lambda^2 - {v_1}^2 - {v_2}^2 - \sqrt{({v_1}^2 + {v_2}^2)^2 + \lambda(-2{v_1}^3 + 6v_1{v_2}^2) + \lambda^2({v_1}^2 + {v_2}^2)}.$$

By noticing that $e_2 > e_3$ and $e_2 e_3 = 2e_1$, we deduce that P is definite positive if $e_3 > 0$. □

Remark 2.6. *The hyperbolicity condition on v_1, v_2 and λ is the same as the diffusive stability condition given in the Proposition 2.3. We have represented it graphically in Figure 6.1. Here again, the diffusive analysis and the hyperbolicity analysis are equivalent.*

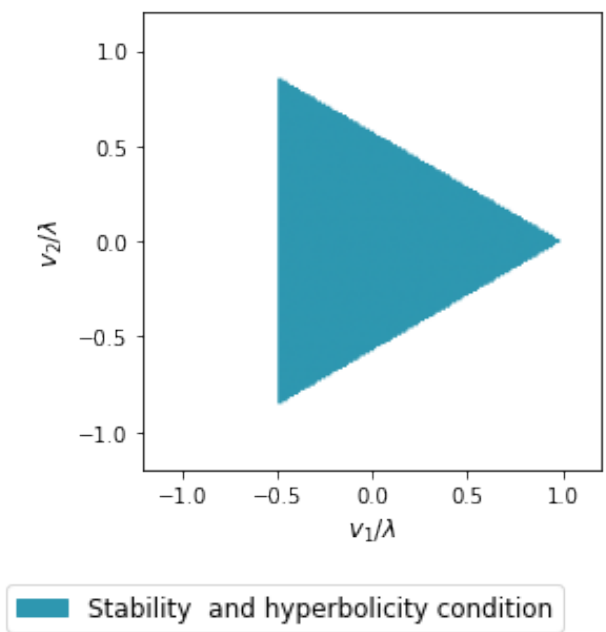

Figure 6.1: Graphic representation of the diffusive stability and hyperbolicity condition of the $D2Q3$ model.

2.4 The $D2Q4$ model

Diffusive stability condition

Proposition 2.5. *The $D2Q4$ model is stable if ${v_1}^2 + {v_2}^2 \leqslant \frac{\lambda^2}{2}$.*

Proof. We have

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = \frac{\Delta t}{2}\left(\frac{1}{\omega} - \frac{1}{2}\right)\nabla \cdot (\mathcal{D}_4 \nabla w) + O(\Delta t), \tag{6.23}$$

with $\mathcal{D}_4 = \begin{pmatrix} \frac{\lambda^2}{2} - {v_1}^2 & -v_1 v_2 \\ -v_1 v_2 & \frac{\lambda^2}{2} - {v_2}^2 \end{pmatrix}$.

The model is stable if the diffusion matrix $\mathcal{D}_4$ is positive. Its eigenvalues are:

$$d_1 = \frac{1}{2}\left(\lambda^2 - \|v\|^2 - \sqrt{(\lambda^2 - \|v\|^2)^2 - \lambda^4 + 2\lambda^2 {v_2}^2 + 2\lambda^2 {v_1}^2}\right) \quad \text{and}$$
$$d_2 = \frac{1}{2}\left(\lambda^2 - \|v\|^2 + \sqrt{(\lambda^2 - \|v\|^2)^2 - \lambda^4 + 2\lambda^2 {v_2}^2 + 2\lambda^2 {v_1}^2}\right).$$

As $d_1 \leqslant d_2$, the eigenvalues are both positive if $d_1 \geqslant 0$, which means if

$${v_1}^2 + {v_2}^2 \leqslant \frac{\lambda^2}{2}.$$

□

Hyperbolicity condition

Proposition 2.6. *The matrix*

$$P = \begin{pmatrix} \lambda^2(4{v_1}^2 - \lambda^2)(4{v_2}^2 - \lambda^2) & 0 & 0 & 0 \\ 0 & -2\lambda^2(4{v_2}^2 - \lambda^2) & 0 & 2v_1(4{v_2}^2 - \lambda^2) \\ 0 & 0 & -2\lambda^2(4{v_1}^2 - \lambda^2) & -2v_2(4{v_1}^2 - \lambda^2) \\ 0 & 2v_1(4{v_2}^2 - \lambda^2) & -2v_2(4{v_1}^2 - \lambda^2) & -2{v_1}^2 - 2{v_2}^2 + \lambda^2 \end{pmatrix},$$

symmetrizes the equivalent system of the D2Q4 model (4.17), if

$$4\max({v_1}^2, {v_2}^2) < \lambda^2. \tag{6.24}$$

Consequently, under this condition, the equivalent system (4.17) is hyperbolic.

Proof. We are searching for a matrix

$$P = \begin{pmatrix} p_1 & p_2 & p_3 & p_4 \\ p_2 & p_5 & p_6 & p_7 \\ p_3 & p_6 & p_8 & p_9 \\ p_4 & p_7 & p_9 & p_{10} \end{pmatrix},$$

such as PB_1 and PB_2 are symmetric and P is symmetric positive definite.

We can compute PB_1 and PB_2. As we want these matrices to be symmetric, we obtain conditions on the coefficients p_i. We deduce that

$$P = \begin{pmatrix} \frac{1}{2v_1}\lambda^2(2v_1 - \lambda)(2v_1 + \lambda)p_7 & 0 & 0 & 0 \\ 0 & -p_7\lambda^2/a & 0 & p_7 \\ 0 & 0 & -\frac{p_7\lambda^2(2v_1-\lambda)(2v_1+\lambda)}{(v_1(2v_2-\lambda)(2v_2+\lambda))} & -\frac{v_2 p_7(2v_1-\lambda)(2v_1+\lambda)}{(v_1(2v_2-\lambda)(2v_2+\lambda))} \\ 0 & p_7 & -\frac{v_2 p_7(2v_1-\lambda)(2v_1+\lambda)}{(v_1(2v_2-\lambda)(2v_2+\lambda))} & -\frac{1(2{v_1}^2+2{v_2}^2-\lambda^2)p_7}{2(v_1(2v_2-\lambda)(2v_2+\lambda))} \end{pmatrix}.$$

By choosing $p_7 = 2v_1(2v_2 - \lambda)(2v_2 + \lambda)$, we obtain

$$P = \begin{pmatrix} \lambda^2(4{v_1}^2 - \lambda^2)(4{v_2}^2 - \lambda^2) & 0 & 0 & 0 \\ 0 & -2\lambda^2(4{v_2}^2 - \lambda^2) & 0 & 2v_1(4{v_2}^2 - \lambda^2) \\ 0 & 0 & -2\lambda^2(4{v_1}^2 - \lambda^2) & -2v_2(4{v_1}^2 - \lambda^2) \\ 0 & 2v_1(4{v_2}^2 - \lambda^2) & -2v_2(4{v_1}^2 - \lambda^2) & -2{v_1}^2 - 2{v_2}^2 + \lambda^2 \end{pmatrix}. \tag{6.25}$$

As P is symmetric, according to the Sylvester's criterion, P is positive definite if and only if all of the leading principal minors are positive, that is to say if the following conditions are satisfied

$$\begin{cases} |P_1| &= \lambda^2(4{v_2}^2 - \lambda^2)(4{v_1}^2 - \lambda^2) &> 0, \\ |P_2| &= -2\lambda^4(4{v_1}^2 - \lambda^2)(4{v_2}^2 - \lambda^2)^2 &> 0, \\ |P_3| &= 4\lambda^6(4{v_1}^2 - \lambda^2)^2(4{v_2}^2 - \lambda^2)^2 &> 0, \\ |P_4| &= 4\lambda^4(4{v_1}^2 - \lambda^2)^3(4{v_2}^2 - \lambda^2)^3 &> 0. \end{cases}$$

This is equivalent to

$$\begin{cases} 4{v_1}^2 &< \lambda^2, \\ 4{v_2}^2 &< \lambda^2. \end{cases}$$

It can also be rewritten

$$4\max\left({v_1}^2, {v_2}^2\right) < \lambda^2,$$

or

$$2\max\left(|v_1|, |v_2|\right) < \lambda.$$

□

Remark 2.7. *The hyperbolicity condition obtained is more restrictive than the diffusive stability condition obtained in Proposition 2.5. We can see in Figure 6.2 the values of v_1/λ and v_2/λ for which the diffusive stability condition is verified, the circle colored in yellow, are included in the blue square, for which the hyperbolicity condition is checked. This is coherent with the review of stability conditions given by Bouchut in [14].*

Numerical comparison of the stability conditions

As we can see in Figure 6.2, for some choice of velocity $\boldsymbol{v} = (v_1, v_2)$ and norm of the kinetic velocity λ, the diffusive stability condition can be satisfied, but not the hyperbolicity condition. We want to test numerically what happened when we are in this case.

We consider a square geometry $[0,1] \times [0,1]$ with periodic boundary conditions. We consider $Nx = 200$ space steps in both directions. We initialized w with a Gaussian function centered in the middle of the square

$$w(\boldsymbol{x}, 0) = e^{-80((x_1-0.5)^2+(x_2-0.5)^2)}.$$

Let us choose $\boldsymbol{v} = (1, 0)$. The stability condition is satisfied if

$$\lambda > \sqrt{2({v_1}^2 + {v_2}^2)} = \sqrt{2}.$$

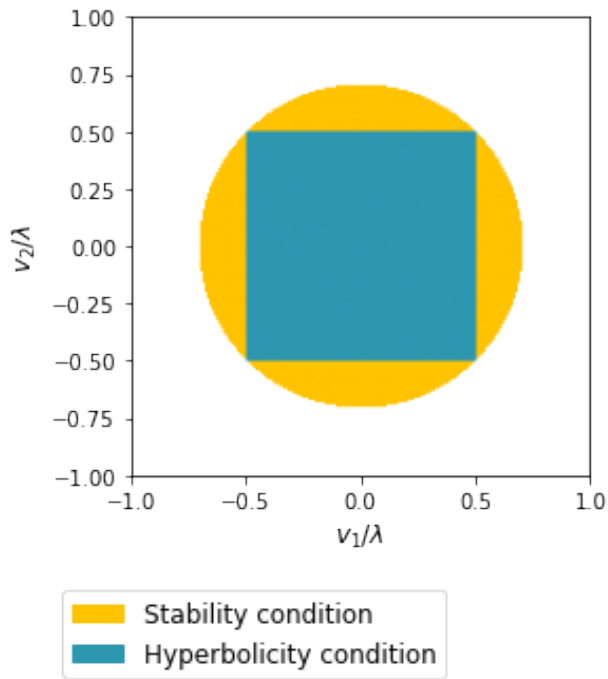

Figure 6.2: Graphic representation of the diffusive stability and hyperbolicity condition of the $D2Q4$ model.

The hyperbolicity condition is satisfied if

$$\lambda > 2\max(|v_1|, |v_2|) = 2.$$

We are going to compare the solution obtained with $\lambda = 1.6$, that is when the diffusive stability condition is satisfied, but not the hyperbolicity condition, and $\lambda = 2.2$, namely when both the diffusive stability and the hyperbolicity conditions are satisfied.

We draw the solutions $w(\boldsymbol{x}, T)$ at time $T = 1$, for different values of the relaxation parameter: $\omega = 2$, $\omega = 1.6$ and $\omega = 2$.

As we are solving the transport step of time step $\frac{\Delta t}{4}$ with a Lattice-Boltzmann method, we need to have the relation between the time and space step

$$\Delta t = \frac{4\Delta x}{\lambda} = \frac{4}{\lambda N x}.$$

Consequently, the number of time step is

$$Nt = \frac{T}{\Delta t} = \frac{\lambda N x}{4},$$

and which depends on the λ chosen: we do $Nt = 80$ time steps when $\lambda = 1.6$ and $Nt = 110$ steps when $\lambda = 2.2$.

We obtained the solution w of the Table 6.1. When $\lambda = 2.2$, that is to say when both the diffusive stability and the hyperbolicity conditions are verified, we obtained a Gaussian

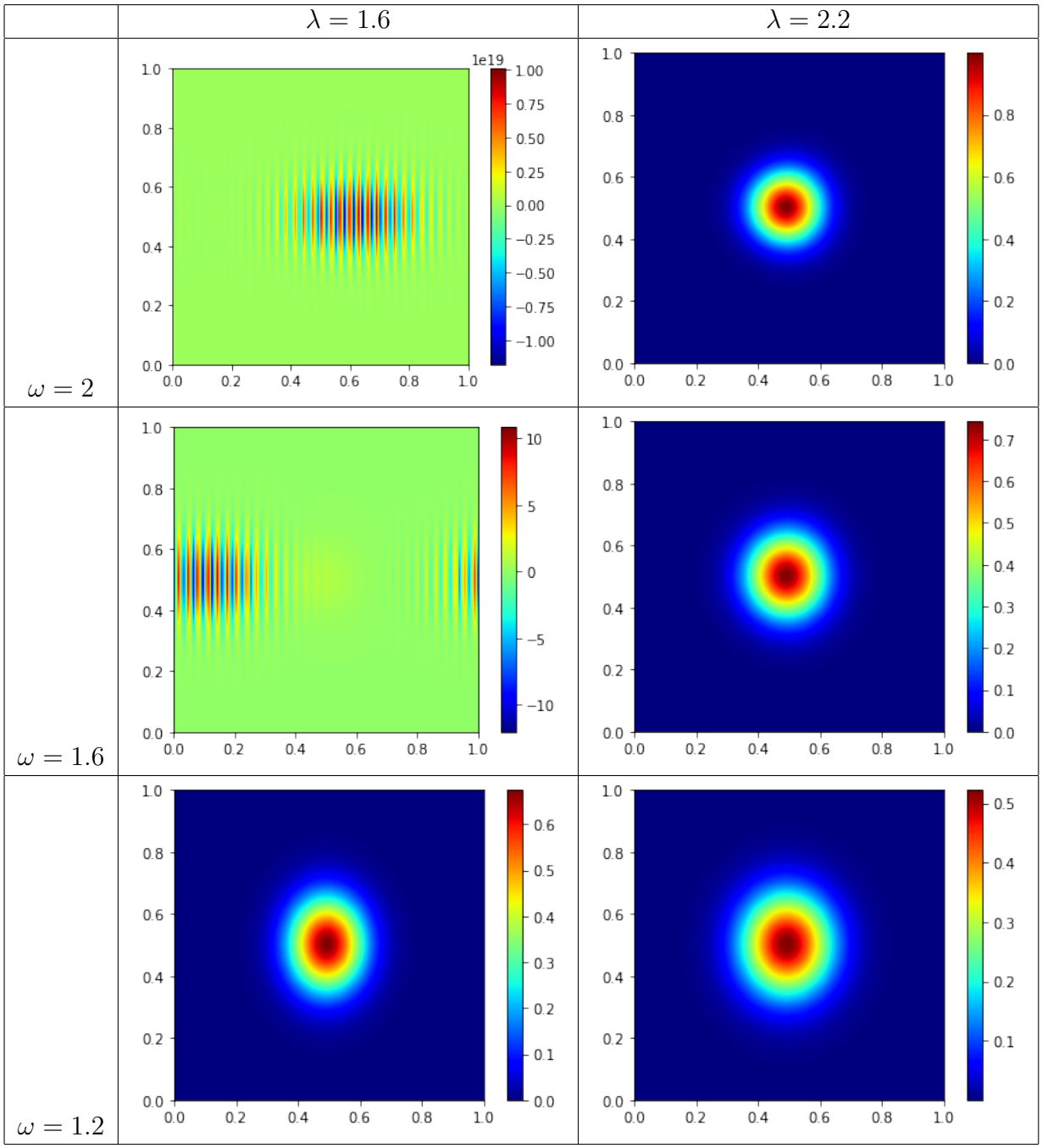

Table 6.1: Solutions w at time $T = 1$ for different values of λ and ω.

centered in the middle of the square, as expected. However, the closer the relaxation parameter ω is to 1, the more the Gaussian function dampens due to the relaxation step. When $\omega = 2$ or $\omega = 1.6$, the solutions obtained with $\lambda = 1.6$, namely when the diffusive stability condition is satisfied but not the hyperbolicity condition, are not stable. Oscillations appear and grow over time. When $\omega = 1.2$ and $\lambda = 1.6$, we obtained a solution close to the expected Gaussian function, but a little distorted. Moreover, this solution is stable, we do not observe any oscillations.

Remark 2.8. *In the Subsection 5, we have shown numerically that the equivalent system is*

a good approximation of the $D1Q2$ model when ω is greater than 1.5 approximately, but not when ω is smaller. Here, although we have used the $D2Q4$ model, we can guess a similar result. The verification of the hyperbolicity condition, obtained with the equivalent system, seems to be necessary to have a stable solution, when ω takes values close to 2. However, for $\omega = 1.2$, we obtained a stable solution even if this hyperbolicity condition is not satisfied.

3 Boundary conditions

We want to solve the transport equation

$$\partial_t w + \boldsymbol{v} \cdot \nabla w = 0.$$

We denote by $s(\boldsymbol{x}, t)$ the exact solution of this equation. We note w_i^n the approximation of $w(i\Delta x, n\Delta t)$ by a kinetic relaxation scheme. The transport steps are solved with a Lattice-Boltzmann Method. To simplify, we consider a regular mesh aligned with the kinetic velocities. We note Ω the geometry considered.

The error between the approximated and the exact solution is defined by

$$e_{Nx} = \frac{\sqrt{\Delta x \sum_{i=1}^{Nx} \left(w_i^{Nt} - s(x_i, T)\right)^2}}{|\Omega|}.$$

Proposition 3.1. *A numerical scheme converges at the order k if, when $Nx \to \infty$,*

$$\frac{e_{Nx}}{e_{2Nx}} \to 2^k.$$

We want to find boundary conditions that give us a second order accuracy and which are stable.

3.1 The $D1Q2$ model

Boundary conditions of order 2

With the $D1Q2$ model defined in Subsection 2.1 (p.17), it is possible to achieve a second order accuracy using the boundary conditions described in [33]. Let us recall these conditions.

We denote by $(f_k)_i^{n+1,*}$ the value of the kth kinetic unknown at the point $x = i\Delta x$ and at time $t = (n+1)\Delta t$ after the transport step, but before the relaxation step. As the transport step is given by a Lattice-Boltzmann method, we have

$$(f_k)_i^{n+1,*} = (f_k)_{i-\frac{\lambda_k}{\lambda}}^n.$$

In each border, there is one transport equation that we cannot solve, because the point $x - \Delta t \lambda_k$ is not included in the mesh. If we note n the outward unit normal vector at point x on the boundary, then we cannot define the kinetic unknown $f_b^*(x, t + \Delta t)$ with $b \in \{1, 2\}$

such as $n \cdot \lambda_b = -\lambda$. For example, when x is located at the left border, we cannot define f_1. At the right border, we cannot define f_2. So we need to add one boundary condition at the border to be able to define the last kinetic unknown. We denote $\bar{b}$ the index of the kinetic velocity opposed to λ_b. We have $n \cdot \lambda_{\bar{b}} = \lambda$. Let us remark that the kinetic velocities of the $D1Q2$ model are defined such as $\lambda_{\bar{b}} = -\lambda_b$.

- On the inflow border, we are able to compute $(f_{\bar{b}})_i^{n+1,*}$, but we need to impose one condition to be able to find $(f_b)_i^{n+1,*}$. We consider a Dirichlet boundary condition on w. For that, we impose w with the exact solution at time $t + \Delta t$

$$w_i^{n+1,*} = s(i\Delta x, t + \Delta t).$$

  Then, as we have

$$w_i^{n+1,*} = \sum_{k=1}^{2} (f_k)_i^{n+1,*},$$

  we obtain

$$(f_b)_i^{n+1,*} = w_i^{n+1,*} - (f_{\bar{b}})_i^{n+1,*}.$$

- On the outflow border, we consider a Neumann condition on y, as proposed in [23]

$$y_i^{n+1,*} = y_{i-1}^{n+1,*}.$$

  Then, as we have

$$\left\{ \begin{array}{rcl} w_i^{n+1,*} & = & (f_b)_i^{n+1,*} + (f_{\bar{b}})_i^{n+1,*}, \\ y_i^{n+1,*} + v w_i^{n+1,*} & = & \lambda_b \left((f_b)_i^{n+1,*} - (f_{\bar{b}})_i^{n+1,*} \right), \end{array} \right.$$

  we obtain the equation

$$y_i^{n+1,*} + v \left((f_b)_i^{n+1,*} + (f_{\bar{b}})_i^{n+1,*} \right) = \lambda_b \left((f_b)_i^{n+1,*} - (f_{\bar{b}})_i^{n+1,*} \right).$$

  Then we can compute the kinetic unknown

$$(f_b)_i^{n+1,*} = \frac{1}{\lambda_b - v} \left(y_i^{n+1,*} + (v + \lambda_b)(f_{\bar{b}})_i^{n+1,*} \right).$$

Numerical tests

Order of convergence

We consider a velocity $v = 0.5$, and the norm of the kinetic velocity $\lambda = 1$, which respect the $D1Q2$ diffusive stability condition defined in Proposition 2.1 and the hyperbolicity condition, defined in Proposition 2.2. We choose a relaxation parameter $\omega = 2$. We consider a one-dimensional space $[0, 1]$. We consider $Nx = 64 \times 2^{i+1}$ space steps and $Nt = 16 \times 2^{i+1}$ time steps, for $i = 1, \ldots, 8$. As we consider the time step $\Delta t = \frac{4}{\lambda Nx}$, the final time $T = Nt\Delta t = 1$ remains identical for all $i = 1, \ldots, 8$.

We initialize w with the compact support function

$$w(x, 0) = \left\{ \begin{array}{ll} 0 & \text{if } r(x) > 1, \\ (1 - r(x)^2)^5 & \text{otherwise,} \end{array} \right.$$

Test case	x_0
1	-0.5
2	0
3	0.5

Table 6.2: Parameters of the test cases for the $D1Q2$ model.

where $r(x) = \frac{|x-x_0|}{\sigma}$, with $\sigma = 0.2$.

We consider 3 different test cases:

(1) The peak starts outside the segment and arrives at the left border.

(2) The peak starts at the left border and arrives in the middle of the segment.

(3) The peak starts in the middle of the segment and arrives at the right border.

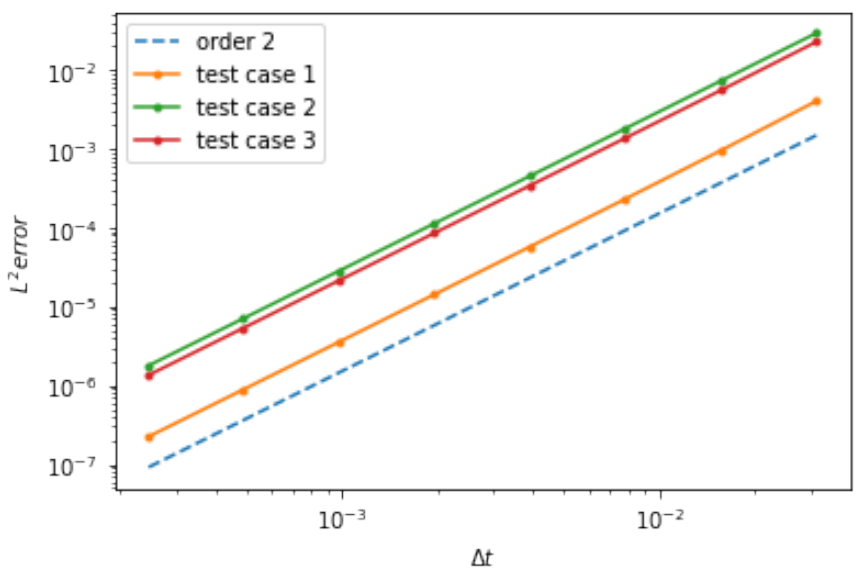

Figure 6.3: L^2 errors between the exact and the numerical solution, for different test cases, with compact support initialization, with a Neumann boundary condition on y at the outflow border.

We obtain the graphics of convergence of Figure 3.1. We can observe a convergence of order 2 for the three test cases.

Stability in a long time

Now, we want to test the stability in a long time. We keep $Nx = 128$ space steps, but we increase the number of time steps: $Nt = 32000$. It gives us a final time $T = 1000$. We take the first test case described previously, and draw the maximum of w for each time step.

We observe in Figure 6.4, that at time $T = 1000$, the sup norm $||w(T)||_\infty$ of w is of order 10^{-6}. We can conclude that these boundary conditions are stable in a long time.

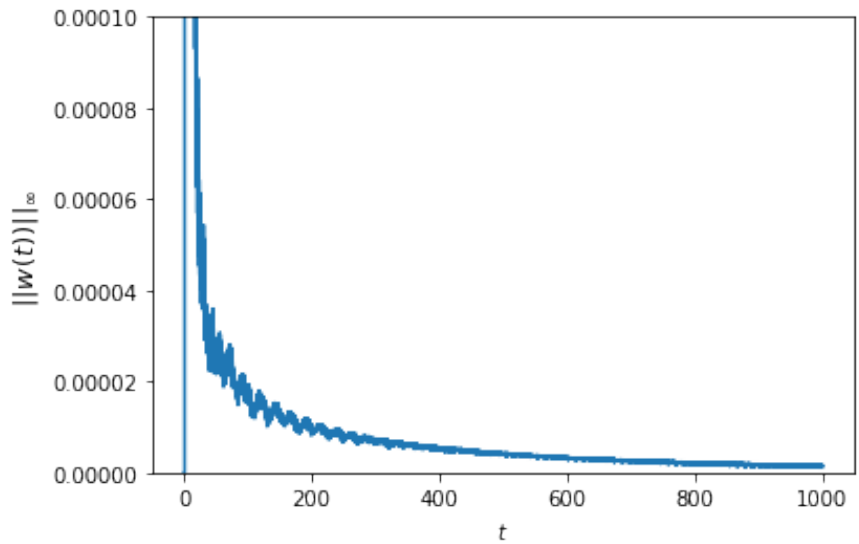

Figure 6.4: Sup norm of the solution w with respect to the time, with a Neumann boundary condition on y at the outflow border.

In conclusion, imposing the exact solution for w at the inflow border and a Neumann condition on y at the outflow border leads to stable boundary conditions of order 2. This result is only a numerical verification. Now, we will provide several tools for analyzing more rigorously the stability of the boundary strategy.

Decrease of the entropy

Let us draw the evolution of the entropy

$$\sum_{i=0}^{Nx+1} w(x_i,t)^2 + \frac{1}{\lambda^2 - v^2} y(x_i,t)^2.$$

We test with the first test case: the peak starts outside the interval and crosses it. We obtain the Figure 6.5. We observe an increase of the entropy at the beginning of the simulation, when the peak enters the interval, and then a decrease until the final time T.

Clearly when the boundary term vanishes, i.e. when the peak does not touch the boundaries anymore, the boundary strategy seems to lead to a scheme that is entropy diminishing.

Stable boundary conditions

Now we propose stable boundary conditions and provide two theoretical explanations of this stability. The first explanation is based on an analysis of the boundary conditions that are stable for the third order equivalent equation. The second explanation relies on an entropy estimate at the boundary.

Stability analysis with the equivalent equation

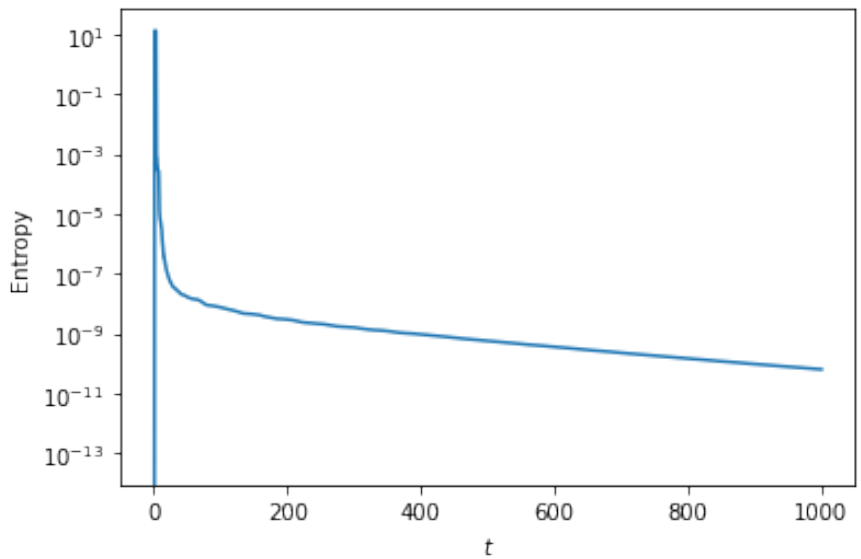

Figure 6.5: Evolution of the entropy with respect to the time (with a log scale for the vertical axis), with a Neumann boundary condition on y at the outflow border.

In chapter 4, we have computed the equivalent equation 4.14 on w of the $D1Q2$ model. We obtained

$$\partial_t w + v\partial_x w + \frac{\Delta t}{2}\left(\frac{1}{2}-\frac{1}{\omega}\right)(\lambda^2 - v^2)\partial_{xx} w + \Delta t^2 \frac{\omega^2 - 6\omega + 6}{12\omega^2}(v^2 - \lambda^2) v \partial_{xxx} w = O(\Delta t^3). \quad (6.26)$$

We consider $\omega = 2$. Let us multiply this equation by w and integrate:

$$\int_{x_{min}}^{x_{max}} w\partial_t w + v\int_{x_{min}}^{x_{max}} w\partial_x w = O(\Delta t^2). \quad (6.27)$$

To have the stability, we need to satisfy

$$v\int_{x_{min}}^{x_{max}} w\partial_x w \geqslant 0. \quad (6.28)$$

By performing integration by parts, we have

$$\frac{v}{2}\left[w^2\right]_{x_{min}}^{x_{max}} = \frac{v}{2}\left(w^2(x_{max}) - w^2(x_{min})\right) \geqslant 0. \quad (6.29)$$

If $v > 0$, we can choose $w(x_{min}) = 0$. If $v < 0$, we can choose $w(x_{max}) = 0$.

Now, let us multiply Equation (6.26) by w and integrate and keep all the terms. We have

$$\int_{x_{min}}^{x_{max}} w\partial_t w + v\int_{x_{min}}^{x_{max}} w\partial_x w + \frac{\Delta t^2}{24}(\lambda^2 - v^2) v \int_{x_{min}}^{x_{max}} w\partial_{xxx} w = O(\Delta t^3). \quad (6.30)$$

To achieve the stability, we can impose

$$\frac{\Delta t^2}{24}(\lambda^2 - v^2) v \left([w\partial_{xx} w]_{x_{min}}^{x_{max}} - \int_{x_{min}}^{x_{max}} \partial_x w \partial_{xx} w\right) \geqslant 0.$$

As the $D1Q2$ stability condition is satisfied, we have $v^2 < \lambda^2$. Let us assume that $v > 0$. Then, to achieve the stability of order 2, we have $w(x_{min}) = 0$. We need to satisfy

$$w(x_{max})\partial_{xx}w(x_{max}) + \frac{1}{2}\left((\partial_x w(x_{min}))^2 - (\partial_x w(x_{max}))^2\right) \geqslant 0.$$

To obtain this inequality, we can choose $\partial_{xx}w(x_{max}) = 0$ and $\partial_x w(x_{max}) = 0$. Moreover, we have seen that to obtain the equivalent equation of order 3, we can assume that y can be written as a linear combination of the space derivatives of w. In Equation (4.13), we obtained that

$$y = \frac{\Delta t}{a} b_{2,1}\partial_x w + \Delta t^2 \left(\frac{c_{2,1}}{a} - \frac{b_{2,1}b_{1,1}}{a^2} + \frac{b_{2,2}b_{2,1}}{a^2}\right)\partial_{xx}w + O(\Delta t^3).$$

Then, choosing $\partial_{xx}w(x_{max}) = 0$ and $\partial_x w(x_{max}) = 0$ is equivalent to choose $y(x_{max}) = 0$.

With a similar computation, we obtain that if $v < 0$, we can choose $y(x_{min}) = 0$.

Consequently, imposing the exact solution for w at the inflow border, and the Dirichlet boundary condition $y = 0$ at the outflow border is supposed to be stable boundary conditions. We will test numerically the stability of these boundary conditions in Section 3.1.

Stability analysis with a boundary entropy estimate

Let us consider a general diagonal quadratic form in w and y:

$$\begin{aligned}\begin{pmatrix} w \\ y \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & \alpha \end{pmatrix} \begin{pmatrix} w \\ y \end{pmatrix} &= w^2 + \alpha y^2, \\ &= (f_1 + f_2)^2 + \alpha\left((\lambda - v)f_1 - (\lambda + v)f_2\right)^2, \\ &= f_1^2 + 2f_1f_2 + f_2^2 + \alpha(\lambda - v)^2 f_1^2 - 2\alpha(\lambda - v)(\lambda + v)f_1f_2 + (\lambda + v)^2 f_2^2.\end{aligned}$$

If we assume that the quadratic form is also diagonal in the f_i variables, this gives

$$2f_1f_2 - 2\alpha(\lambda - v)(\lambda + v)f_1f_2 = 0.$$

It implies

$$\alpha = \frac{1}{\lambda^2 - v^2}.$$

It gives us the entropy

$$\begin{aligned}\Sigma &= w^2 + \alpha y^2, \\ &= \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \cdot \begin{pmatrix} \frac{2\lambda}{\lambda+v} & 0 \\ 0 & \frac{2\lambda}{\lambda-v} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}.\end{aligned}$$

When we are at the border, the kinetic unknown $f_{\bar{b}}$ is leaving the geometry, while f_b is entering. The LBM algorithm changes the entropy according to

$$\Sigma \leftarrow \Sigma + 2\lambda^2 \left(\frac{1}{\lambda^2 + \lambda_b v} f_b^2 - \frac{1}{\lambda^2 - \lambda_b v} f_{\bar{b}}^2\right).$$

In order to obtain a stable solution, we want the entropy to decrease. In other words, we do not want the density which is entering the geometry to be greater than the one which is leaving it. Consequently, we want

$$\frac{1}{\lambda^2+\lambda_b v} f_b^2 - \frac{1}{\lambda^2-\lambda_b v} f_{\bar{b}}^2 \leqslant 0.$$

It gives us a stability condition on the boundary conditions

$$|f_b| \leqslant \sqrt{\frac{\lambda^2+\lambda_b v}{\lambda^2-\lambda_b v}} |f_{\bar{b}}|. \tag{6.31}$$

Moreover, the kinetic unknowns are defined by

$$f_k = \frac{\lambda^2+\lambda_k v}{2\lambda^2} w + \frac{\lambda_k y}{2\lambda^2}.$$

At an inflow border

We have

$$f_b + f_{\bar{b}} = w.$$

If we choose a Dirichlet boundary condition $w = 0$, then we have

$$f_b = -f_{\bar{b}}.$$

By inserting this expression in (6.31), we obtain

$$\sqrt{\frac{\lambda^2+\lambda_b v}{\lambda^2-\lambda_b v}} \geqslant 1.$$

This equality is true if

$$\lambda_b v \geqslant 0.$$

As we are in an inflow border, we have

$$nv \geqslant 0.$$

Moreover, by definition of b, we have $n\lambda_b = -\lambda$, which gives us

$$n = \frac{-\lambda}{\lambda_b}.$$

We obtain that $\lambda_b v \geqslant 0$. The condition of stability on the boundary is satisfied. In conclusion, imposing the boundary condition $w = 0$ at an inflow border gives us a stable solution.

At an outflow border

We have

$$f_b - \frac{\lambda^2 + \lambda_b v}{\lambda^2 - \lambda_b v} f_{\bar{b}} = \left(1 + \frac{\lambda^2 + \lambda_b v}{\lambda^2 - \lambda_b v}\right) \frac{\lambda_b}{2\lambda^2} y,$$
$$= \frac{\lambda_b}{\lambda^2 - \lambda_b v} y.$$

If we choose to impose a Dirichlet boundary condition on the flux error $y = 0$, we obtain

$$f_b = \frac{\lambda^2 + \lambda_b v}{\lambda^2 - \lambda_b v} f_{\bar{b}}.$$

Inserting this expression in (6.31) gives us

$$\left| \frac{\lambda^2 + \lambda_b v}{\lambda^2 - \lambda_b v} \right| \leqslant \sqrt{\frac{\lambda^2 + \lambda_b v}{\lambda^2 - \lambda_b v}},$$

which is equivalent to

$$\lambda_b v \leqslant 0.$$

As we are in an outflow border, this condition is satisfied. Indeed, we have

$$nv = -\frac{\lambda}{\lambda_b} v > 0.$$

We can conclude that imposing a Dirichlet boundary condition on the flux error $y = 0$ on an outflow border satisfy the decrease of the entropy.

In conclusion, if we impose w with the exact solution on the inflow border and $y = 0$ at the outflow border, we obtain stable boundary conditions. We retrieve the same boundary conditions as in Section 3.1.

Moreover, we have defined a condition (6.31) to ensure the decrease of the entropy and therefore the stability. Let us remark that inequalities on the entropy have already been used to define boundary conditions in [39, 15].

Numerical tests

Order of convergence

Now, let us test numerically the order of convergence and the stability in a long time of the previous boundary conditions. We reuse the test cases described in Section 3.1.

We obtain the graphics of convergence of Figure 3.1. We can observe that only the first two test cases converge at the order 2. The third one converges at the order 1.

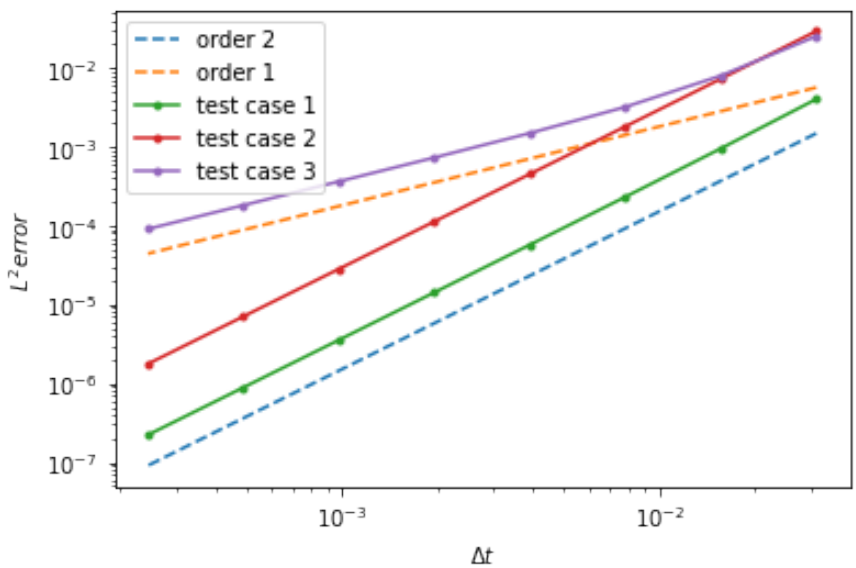

Figure 6.6: L^2 errors between the exact and the numerical solution, for different test cases, with compact support initialization, with a Dirichlet boundary condition on y at the outflow border.

Stability in a long time

Now, we want to test the long time stability. We keep $Nx = 128$ grid points, but we increase the number of time steps: $Nt = 32000$. It gives us a final time $T = 1000$. We take the first test case described previously, and draw the maximum of w for each time step.

We observe in Figure 6.7, that the solution at time $T = 1000$, which is supposed to be null, is of order 10^{-6}. We can conclude that these boundary conditions are stable in a long time.

In conclusion, we are imposing the exact solution for w at the inflow border and a Dirichlet condition $y = 0$ at the outflow border are boundary conditions which are stable in a long time, as we have proven in Section 3.1. However, the solution obtained with these boundary conditions do not always achieve a convergence of order 2.

Decrease of the entropy

As for the first boundary conditions, we can draw the evolution of the entropy

$$\sum_{i=0}^{Nx+1} w(x_i,t)^2 + \frac{1}{\lambda^2 - v^2} y(x_i,t)^2.$$

We obtain the Figure 6.8. We observe an increase of the entropy at the beginning of the simulation, when the peak enters the interval, and then, as expected, a decrease until the final time T.

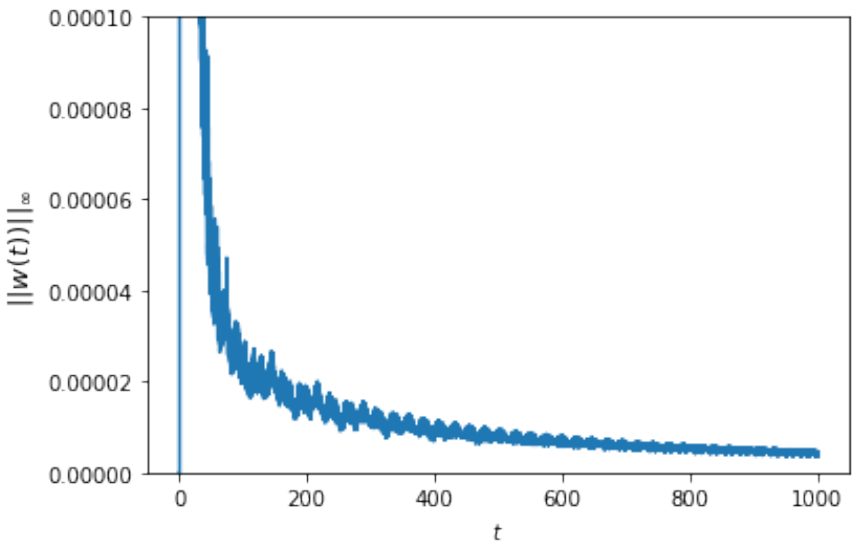

Figure 6.7: Sup norm of the solution w with respect to the time, with a Dirichlet boundary condition on y at the outflow border.

3.2 The $D2Q4$ model

We now consider the $D2Q4$ model, defined in Subsection 2.3 (p.18). We denote the velocity $\boldsymbol{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$.

First, we solve the transport equations (2.10) with respect to each kinetic velocity. Numerically, we do the translation

$$(f_k)_{i,j}^{n+1,*} = (f_k)^n_{i-\frac{\lambda_k^1}{\lambda}, j-\frac{\lambda_k^2}{\lambda}}.$$

In each border, there is one transport equation that we cannot solve, because the point $\boldsymbol{x} - \Delta t \boldsymbol{\lambda}_k$ is not included in the mesh. If we note $\boldsymbol{n} = (n_1, n_2)$ the outward normal vector at point $\boldsymbol{x}$ on the boundary, then we cannot define the kinetic unknown $f_b^*(\boldsymbol{x}, t + \Delta t)$ with $b \in \{1, 2, 3, 4\}$ such as $\boldsymbol{n} \cdot \boldsymbol{\lambda}_b = -\lambda$. We call this border b. For example, when $\boldsymbol{x}$ is located at the left border, we cannot define f_1: thus, the left border is the border 1. So we need to add one boundary condition at the border to be able to define the last kinetic unknown. We denote $\overline{b}$ the index of the kinetic velocity opposed to $\boldsymbol{\lambda_b}$. We have $\boldsymbol{n} \cdot \boldsymbol{\lambda}_{\overline{b}} = \lambda$. Let us remark that the kinetic velocities of the $D2Q4$ model are defined such as $\boldsymbol{\lambda_{\overline{b}}} = -\boldsymbol{\lambda_b}$.

Moreover, let us notice that, as the outward normal vector $\boldsymbol{n}$ is aligned to the cartesian mesh, one of its components is null. On the other hand, all the kinetic velocities of the $D2Q4$ model has one coefficient which is null. Therefore, $\boldsymbol{n} \cdot \boldsymbol{\lambda}_b = -\lambda$ implies that we have necessarily

$$\boldsymbol{n} = -\frac{\boldsymbol{\lambda}_b}{\lambda}. \tag{6.32}$$

When $\boldsymbol{x}$ is located in a corner, that is at the intersection of two borders, then, there are two kinetic unknowns that we cannot define by solving the transport equations. At these points, we need to impose two boundary conditions.

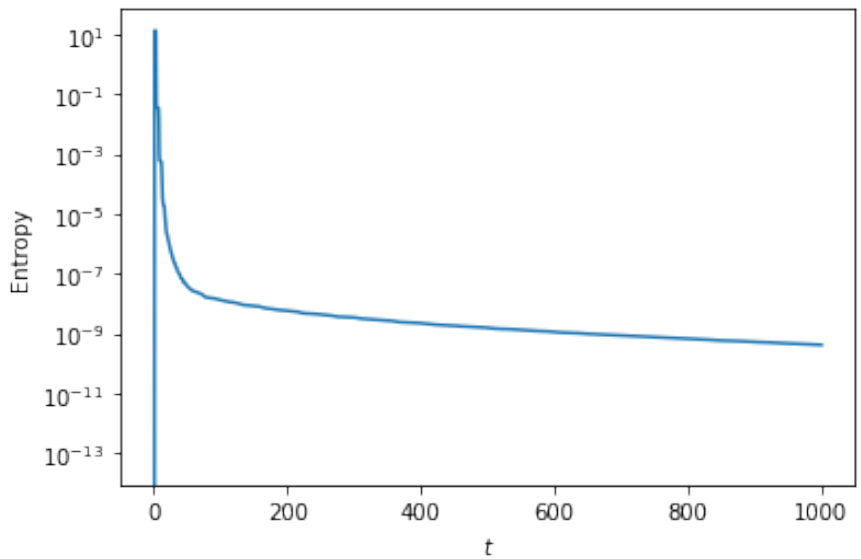

Figure 6.8: Evolution of the entropy with respect to the time (with a log scale for the vertical axis), with a Dirichlet boundary condition on y at the outflow border.

We note b_1 and b_2 the indices of the two borders that form the corner, meaning we want to find the value of f_{b_1} and f_{b_2}. To simplify the calculation later, we choose that $b_1 < b_2$. Let us remark that, as a corner is the intersection of a horizontal and a vertical border, we have necessarily $b_1 \in \{1, 2\}$ a horizontal border and $b_2 \in \{3, 4\}$ a vertical border. We note $\overline{b_1}$ and $\overline{b_2}$ the other horizontal and vertical borders, which do not intersect the corner.

Stable boundary condition

Using the moment matrix of the $D2Q4$ model (2.26) and the definition of the flux errors (3.3), we obtain the expression of Y from the kinetic unknowns F

$$Y = MF, \tag{6.33}$$

with

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 \\ \lambda - v_1 & -\lambda - v_1 & -v_1 & -v_1 \\ -v_2 & -v_2 & \lambda - v_2 & -\lambda - v_2 \\ \lambda^2 & \lambda^2 & -\lambda^2 & -\lambda^2 \end{pmatrix}.$$

With the matrix P defined by (6.25) which symmetries the equivalent system (4.17), we can define a quadratic form on Y

$$Y \cdot PY = Y^\top PY.$$

Using the change of variables (6.33), we obtain

$$\begin{aligned} Y^\top PY &= (MF)^\top PMF, \\ &= F^\top M^\top PMF, \end{aligned}$$

where

$$M^\top PM = 4\lambda^3(\lambda^2 - 4{v_2}^2)(\lambda^2 - 4{v_1}^2)\begin{pmatrix} \frac{1}{\lambda+2v_1} & 0 & 0 & 0 \\ 0 & \frac{1}{\lambda-2v_1} & 0 & 0 \\ 0 & 0 & \frac{1}{\lambda+2v_2} & 0 \\ 0 & 0 & 0 & \frac{1}{\lambda-2v_2} \end{pmatrix}.$$

We obtain the entropy on the kinetic unknown $\boldsymbol{f}$

$$\Sigma(\boldsymbol{f}) = 4\lambda^3(\lambda^2 - 4{v_2}^2)(\lambda^2 - 4{v_1}^2)\left(\frac{1}{\lambda+2v_1}f_1^2 + \frac{1}{\lambda-2v_1}f_2^2 + \frac{1}{\lambda+2v_2}f_3^2 + \frac{1}{\lambda-2v_2}f_4^2\right).$$

Let $\boldsymbol{x}$ be a point located at the border b, but not in a corner. Then, the value of f_b is missing, and we need to define its value.

At this point $\boldsymbol{x}$, f_b is entering inside the geometry while $f_{\bar{b}}$ is leaving outside. Then, the entropy is given by

$$\Sigma(\boldsymbol{x}, t+\Delta t) = \Sigma(\boldsymbol{x}, t) + 4\lambda^3(\lambda^2 - 4{v_2}^2)(\lambda^2 - 4{v_1}^2)\left(\frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_b\cdot\boldsymbol{v}}f_b^2 - \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}}\cdot\boldsymbol{v}}f_{\bar{b}}^2\right).$$

To be stable, we need the entropy to decrease with time. In other words, we do not want the density which is entering into the geometry to be greater than the one which is leaving it. Moreover, as $\lambda > 0$ and the hyperbolicity condition of Proposition 2.6 is satisfied, meaning $4\max({v_1}^2, {v_2}^2) < \lambda^2$, we want to satisfy

$$\frac{1}{\lambda^2 + 2\boldsymbol{\lambda}_b\cdot\boldsymbol{v}}f_b^2 - \frac{1}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}}\cdot\boldsymbol{v}}f_{\bar{b}}^2 \leqslant 0,$$

which gives us the decreasing entropy condition at the border b

$$|f_b| \leqslant \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_b\cdot\boldsymbol{v}}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}}\cdot\boldsymbol{v}}}|f_{\bar{b}}|. \tag{6.34}$$

The kinetic unknown f_i are given by inverting the matrix M. We have

$$f_k = \frac{\lambda^2 + 2\boldsymbol{\lambda}_k\cdot\boldsymbol{v}}{4\lambda^2}w + \frac{\boldsymbol{\lambda}_k\cdot\begin{pmatrix} y_1 \\ y_2 \end{pmatrix}}{2\lambda^2} + \frac{((\lambda_k^1)^2 - (\lambda_k^2)^2)\,z_3}{4\lambda^4}.$$

Boundary conditions on inflow borders

If we are in an inflow border b, as $\lambda_b = -\lambda_{\bar{b}}$, we have

$$f_b + f_{\bar{b}} = \frac{w}{2} + \frac{((\lambda_b^1)^2 - (\lambda_b^2)^2)\,z_3}{2\lambda^4}.$$

If we impose $w = 0$ and $z_3 = 0$, then, we have

$$f_b = -f_{\bar{b}}.$$

If we insert this equality in (6.34), we obtain

$$\begin{aligned} & \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}} \cdot \boldsymbol{v}}} \geqslant 1, \\ \iff \quad & \lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v} \geqslant \lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}, \\ \iff \quad & \boldsymbol{\lambda}_b \cdot \boldsymbol{v} \geqslant 0. \end{aligned}$$

As we are at an inflow boundary, we have

$$\boldsymbol{n} \cdot \boldsymbol{v} \leqslant 0,$$

with $\boldsymbol{n}$ an outward normal vector of this border. We remind that we have (6.32)

$$\boldsymbol{n} = -\frac{\boldsymbol{\lambda}_b}{\lambda}.$$

We can conclude that the inequality $\boldsymbol{\lambda}_b \cdot \boldsymbol{v} \geqslant 0$ is satisfied. The boundary condition $w = 0$ and z_3 in inflow borders satisfies the decrease of the entropy.

Boundary conditions on outflow borders

Let us consider an outflow border b. We have

$$\begin{aligned} f_b - \frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}} f_{\bar{b}} &= \frac{\boldsymbol{\lambda}_b \cdot \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}}{2\lambda^2} \left(\frac{2\lambda^2}{\lambda^2 - 2\boldsymbol{\lambda}_k \cdot \boldsymbol{v}} \right) + \frac{((\lambda_b^1)^2 - (\lambda_b^2)^2)\, z_3}{4\lambda^4} \left(1 - \frac{\lambda^2 + 2\boldsymbol{\lambda}_k \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_k \cdot \boldsymbol{v}} \right), \\ &= \frac{\boldsymbol{\lambda}_b \cdot \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}}{\lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}} - \frac{((\lambda_b^1)^2 - (\lambda_b^2)^2)}{\lambda^4} \frac{\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_k \cdot \boldsymbol{v}} z_3. \end{aligned}$$

Let us denote $y_c = \frac{\lambda_b^1 y_1 + \lambda_b^2 y_2}{\lambda}$, the component of the flux error which follows the direction of the normal vector n. If we impose the Dirichlet boundary conditions $y_c = 0$ and $z_3 = 0$ then we have

$$f_b = \frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}} f_{\bar{b}}.$$

By inserting this expression of f_b in (6.34), we obtain

$$\begin{aligned} & \left| \frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}} \right| \leqslant \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}}, \\ \iff \quad & \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}} \leqslant 1, \\ \iff \quad & \boldsymbol{\lambda}_b \cdot \boldsymbol{v} \leqslant 0. \end{aligned}$$

As b is an outflow border, we have

$$\boldsymbol{n} \cdot \boldsymbol{v} > 0.$$

Then, using the equality (6.32) that $\boldsymbol{n} = -\frac{\boldsymbol{\lambda}_b}{\lambda}$, the inequality $\boldsymbol{\lambda}_b \cdot \boldsymbol{v} \leqslant 0$ is satisfied. In conclusion, imposing the component of the flux error $y_c = 0$ which follow the direction of the normal vector $\boldsymbol{n}$ and $z_3 = 0$ on the outflow borders gives us stable boundary conditions.

Boundary conditions at the corners

Let us note b_1 and b_2 the two borders that form the corner, such as $b_1 < b_2$.

In this corner $\boldsymbol{x}$, the difference of entropy between two time steps is

$$\Sigma(\boldsymbol{x}, t+\Delta t) = \Sigma(\boldsymbol{x}, t) + 4\lambda^3(\lambda^2 - 4{v_2}^2)(\lambda^2 - 4{v_1}^2) \\ \left(\frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{b_1}\cdot\boldsymbol{v}} f_{b_1}^2 - \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}_1}\cdot\boldsymbol{v}} f_{\bar{b}_1}^2 + \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{b_2}\cdot\boldsymbol{v}} f_{b_2}^2 - \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}_2}\cdot\boldsymbol{v}} f_{\bar{b}_2}^2\right).$$

As previously, we want the entropy to decrease, which means

$$\frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{b_1}\cdot\boldsymbol{v}} f_{b_1}^2 - \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}_1}\cdot\boldsymbol{v}} f_{\bar{b}_1}^2 + \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{b_2}\cdot\boldsymbol{v}} f_{b_2}^2 - \frac{\lambda}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}_2}\cdot\boldsymbol{v}} f_{\bar{b}_2}^2 \leqslant 0.$$

One sufficient condition to achieve that is to satisfy the two inequalities

$$|f_{b_1}| \leqslant \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_{b_1}\cdot\boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_{b_1}\cdot\boldsymbol{v}}}|f_{\bar{b}_1}| \qquad \text{and} \qquad |f_{b_2}| \leqslant \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_{b_2}\cdot\boldsymbol{v}}{\lambda^2 - 2\boldsymbol{\lambda}_{b_2}\cdot\boldsymbol{v}}}|f_{\bar{b}_2}|. \tag{6.35}$$

In the stable boundary conditions described previously for the inflow and the outflow borders, f_b are defined only from $f_{\bar{b}}$. The two other kinetic unknowns do not interfere in the computation.

Therefore, in the corners, we can define f_{b_1} and f_{b_2} from $f_{\bar{b}_1}$ and respectively $f_{\bar{b}_2}$ with the inflow or outflow boundary condition in function to the border.

Numerical tests

Order of convergence

We choose a square geometry, aligned with the kinetic velocities: $\Omega = [0,1]\times[0,1]$. We initialize w with a function with compact support

$$w(x_1, x_2, t) = \begin{cases} 0 & \text{if } r(x_1,x_2) > 1, \\ (1 - r(x_1,x_2)^2)^5 & \text{otherwise.} \end{cases}$$

with $r(x_1,x_2) = \frac{\sqrt{(x_1-x_1^0)^2+(x_2-x_2^0)^2}}{\sigma}$ and $\sigma = 0.4$.

We consider the 7 following test cases, with the parameters of Table 3.2:

(1) The peak starts in the middle of the square and goes toward the right border.

(2) The peak starts outside the square and arrives at the middle of the left border.

(3) The peak starts at the middle of the square and arrives at the middle of the left border.

(4) The peak starts at the middle of the left border and arrives at the middle of the square.

(5) The peak starts outside the square and arrives at the left bottom left corner.

Test case	x_1^0	x_2^0	v_1	v_2
1	0.5	0.5	0.8	0
2	-0.4	0.5	0.8	0
3	0.4	0.5	-0.8	0
4	0	0.5	0.8	0
5	$-\sqrt{2}/4$	$-\sqrt{2}/4$	$\sqrt{2}/2$	$\sqrt{2}/2$
6	$\sqrt{2}/4$	$\sqrt{2}/4$	$-\sqrt{2}/2$	$-\sqrt{2}/2$
7	0	0	$\sqrt{2}/2$	$\sqrt{2}/2$

Table 6.3: Parameters of the test cases for the $D2Q4$ model.

(6) The peak starts at the middle of the square and arrives at the left bottom left corner.

(7) The peak starts at the left bottom left corner and arrives at the middle of the square.

For all these test cases, we choose $\lambda = 2$, which satisfy the hyperbolicity condition (6.24). We choose a final time $T = 0.5$, and a number of time steps $Nt = 16, 32, 64, 128$.

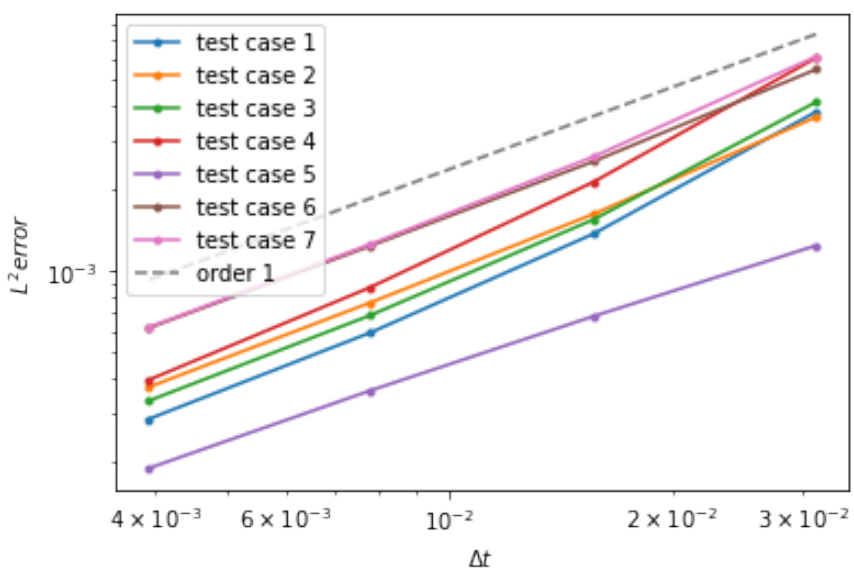

Figure 6.9: L^2 errors between the exact and the numerical solution, for the test cases defined in Table 3.2, with the entropy decreasing boundary conditions.

We obtain Figure 6.9. We can observe that the test cases converge at order 1.

Stability in a long time

If we compute the test case 1 with an increase of the number of time steps to $Nt = 640$, to achieve the final time $T = 20$, we obtain the evolution of the maximum of w with respect to the time in Figure 6.10. At time $T = 20$, we obtain a maximum of order 10^{-5}.

Decrease of the entropy

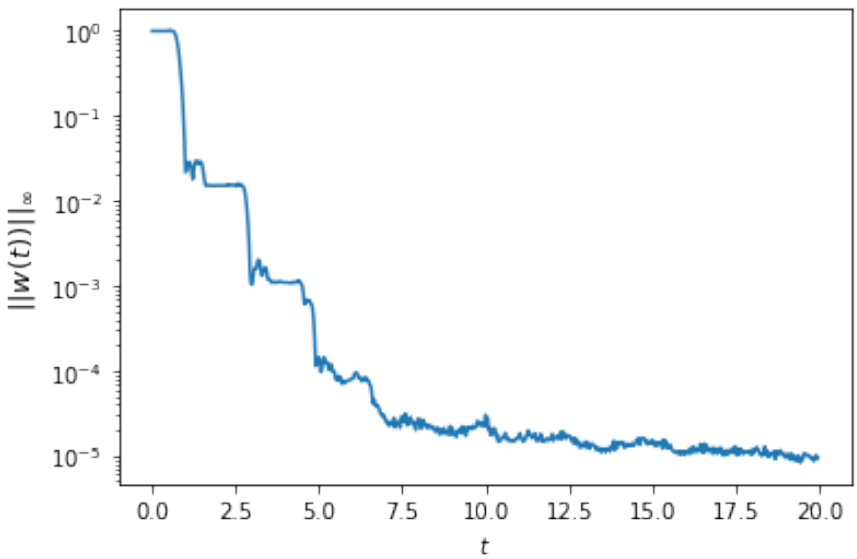

Figure 6.10: Sup norm of the solution w with respect to the time (with a log scale for the vertical axis), for the test case 1 defined in Table 3.2, with the entropy decreasing boundary conditions.

We can also draw the entropy with respect to the time

$$\sum_{x_{i,j}} \boldsymbol{Y}(x_{i,j}, t) \cdot P\boldsymbol{Y}(x_{i,j}, t),$$

with P defined by (6.25), for the test case 1.

We obtain the figure 6.11. We can see that the entropy is decreasing quickly in the beginning of the simulation when the peak is leaving the square geometry. Then, as expected, the numerical entropy continues to decrease with time.

Boundary conditions of order 2

We would like to have boundary conditions of order 2. After many attempts, we proposed below boundary conditions which give us numerically a convergence of order 2.

Boundary conditions on inflow borders

At the inflow borders, namely when $\boldsymbol{v} \cdot \boldsymbol{n} \leqslant 0$, we impose w with a Dirichlet boundary condition at the exact solution at time $t + \Delta t$ and in the point $\boldsymbol{x}$

$$w_{i,j}^{n+1,*} = s((i\Delta x_1, j\Delta x_2), t + \Delta t).$$

Then, the kinetic unknown that we need to define can be written

$$(f_b)_{i,j}^{n+1,*} = w_{i,j}^{n+1,*} - \sum_{\substack{k=1 \\ k \neq b}}^{4} (f_k)_{i,j}^{n+1,*}.$$

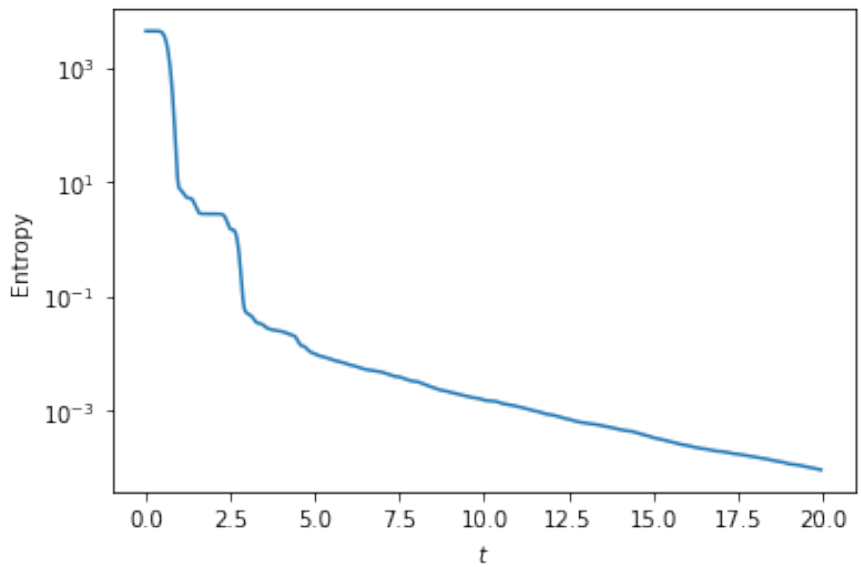

Figure 6.11: Evolution of the entropy with respect to the time (with a log scale for the vertical axis), for the test case 1 defined in Table 3.2, with the entropy decreasing boundary conditions.

Boundary conditions on outflow borders

At the outflow borders, namely when $\boldsymbol{v} \cdot \boldsymbol{n} > 0$, we impose $v_1 y_1 + v_2 y_2$ with a Neumann boundary condition. We take the value of y_1 and y_2 obtained after the transport step and in the point $\boldsymbol{x} = ((i - n_1)\Delta x_1, (j - n_2)\Delta x_2)$ located inside the mesh, not at the border

$$(v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} = v_1 (y_1)_{i-n_1,j-n_2}^{n+1,*} + v_2 (y_2)_{i-n_1,j-n_2}^{n+1,*}.$$

Then, as we have

$$\begin{cases} w_{i,j}^{n+1,*} &= \displaystyle\sum_{k=1}^{4} (f_k)_{i,j}^{n+1,*}, \\ (y_1)_{i,j}^{n+1,*} + v_1 w_{i,j}^{n+1,*} &= \displaystyle\sum_{k=1}^{4} \lambda_k^1 (f_k)_{i,j}^{n+1,*}, \\ (y_2)_{i,j}^{n+1,*} + v_2 w_{i,j}^{n+1,*} &= \displaystyle\sum_{k=1}^{4} \lambda_k^2 (f_k)_{i,j}^{n+1,*}, \end{cases}$$

we need to solve the equation

$$(v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} = v_1 \sum_{k=1}^{4} \lambda_k^1 (f_k)_{i,j}^{n+1,*} + v_2 \sum_{k=1}^{4} \lambda_k^2 (f_k)_{i,j}^{n+1,*} - \left({v_1}^2 + {v_2}^2\right) \sum_{k=1}^{4} (f_k)_{i,j}^{n+1,*}.$$

We obtain

$$(f_b)_{i,j}^{n+1,*} = \frac{1}{-v_1\lambda_b^1 - v_2\lambda_b^2 + {v_1}^2 + {v_2}^2}\Bigg(v_1 \sum_{\substack{k=1\\k\neq b}}^{4} \lambda_k^1 (f_k)_{i,j}^{n+1,*} + v_2 \sum_{\substack{k=1\\k\neq b}}^{4} \lambda_k^2 (f_k)_{i,j}^{n+1,*}$$
$$- \left({v_1}^2 + {v_2}^2\right) \sum_{\substack{k=1\\k\neq b}}^{4} (f_k)_{i,j}^{n+1,*} - (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} \Bigg).$$

Boundary conditions at the corners

Now that we have defined the boundary conditions on the borders, we need to define them on the corners. We remind that at the corners, we need to impose two boundary conditions. We have three possible kinds of corners: the intersection of an inflow border and an outflow border, the intersection of two inflow borders and the intersection of two outflow borders.

Intersection of an inflow and an outflow border

At the intersection of an inflow border and an outflow border, we impose a Dirichlet boundary condition on w

$$w_{i,j}^{n+1,*} = s((i\Delta x_1, j\Delta x_2), t + \Delta t),$$

and a Neumann boundary condition on $v_1 y_1 + v_2 y_2$

$$(v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} = v_1 (y_1)_{i-n_1,j-n_2}^{n+1,*} + v_2 (y_2)_{i-n_1,j-n_2}^{n+1,*}.$$

We then have to solve

$$\begin{cases} w_{i,j}^{n+1,*} &= \displaystyle\sum_{k=1}^{4} (f_k)_{i,j}^{n+1,*}, \\ (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} &= v_1 \sum_{k=1}^{4} \lambda_k^1 (f_k)_{i,j}^{n+1,*} + v_2 \sum_{k=1}^{4} \lambda_k^2 (f_k)_{i,j}^{n+1,*} - \left({v_1}^2 + {v_2}^2\right) w_{i,j}^{n+1,*}. \end{cases}$$

Using the fact that $b_1 \in \{1,2\}$ and $b_2 \in \{3,4\}$, and the expression of the kinetic velocities of the $D2Q4$ model defined in Subsection 2.3 (p.18), we have

$$\begin{cases} (f_{b_1})_{i,j}^{n+1,*} + (f_{b_2})_{i,j}^{n+1,*} &= w_{i,j}^{n+1,*} - (f_{\overline{b_1}})_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*}, \\ v_1 \lambda_{b_1}^1 (f_{b_1})_{i,j}^{n+1,*} + v_2 \lambda_{b_2}^2 (f_{b_2})_{i,j}^{n+1,*} &= -v_1 \lambda_{\overline{b_1}}^1 (f_{\overline{b_1}})_{i,j}^{n+1,*} - v_2 \lambda_{\overline{b_2}}^2 (f_{\overline{b_2}})_{i,j}^{n+1,*} \\ & \quad + \left({v_1}^2 + {v_2}^2\right) w_{i,j}^{n+1,*} + (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*}. \end{cases}$$

We obtain the solution

$$\begin{cases} (f_{b_1})_{i,j}^{n+1,*} &= \frac{1}{-v_1\lambda_{b_1}^1 + v_2\lambda_{b_2}^2} \Bigg(v_2 \lambda_{b_2}^2 \left(w_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*} - (f_{\overline{b_1}})_{i,j}^{n+1,*} \right) \\ & \quad + v_1 \lambda_{\overline{b_1}}^1 (f_{\overline{b_1}})_{i,j}^{n+1,*} + v_2 \lambda_{\overline{b_2}}^2 (f_{\overline{b_2}})_{i,j}^{n+1,*} - \left({v_1}^2 + {v_2}^2\right) w_{i,j}^{n+1,*} - (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} \Bigg), \\ (f_{b_2})_{i,j}^{n+1,*} &= \frac{1}{-v_1\lambda_{b_1}^1 + v_2\lambda_{b_2}^2} \Bigg(- v_1 \lambda_{b_1}^1 \left(w_{i,j}^{n+1,*} - (f_{\overline{b_1}})_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*} \right) \\ & \quad - v_1 \lambda_{\overline{b_1}}^1 (f_{\overline{b_1}})_{i,j}^{n+1,*} - v_2 \lambda_{\overline{b_2}}^2 (f_{\overline{b_2}})_{i,j}^{n+1,*} + \left({v_1}^2 + {v_2}^2\right) w_{i,j}^{n+1,*} + (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} \Bigg). \end{cases}$$

Intersection of two inflow borders

At the intersection of two inflow borders, we impose w with a Dirichlet boundary condition

$$w_{i,j}^{n+1,*} = s((i\Delta x_1, j\Delta x_2), t + \Delta t),$$

and a Dirichlet boundary condition on z_3

$$(z_3)_{i,j}^{n+1,*} = 0.$$

We need to solve the system

$$\left\{ \begin{array}{rcl} w_{i,j}^{n+1,*} & = & \displaystyle\sum_{k=1}^{4} (f_k)_{i,j}^{n+1,*}, \\ (z_3)_{i,j}^{n+1,*} & = & \displaystyle\sum_{k=1}^{4} \left((\lambda_k^1)^2 - (\lambda_k^2)^2\right) (f_k)_{i,j}^{n+1,*}. \end{array} \right.$$

The solution is given by

$$\left\{ \begin{array}{rcl} (f_{b_1})_{i,j}^{n+1,*} & = & \frac{1}{2} w_{i,j}^{n+1,*} - (f_{\overline{b_1}})_{i,j}^{n+1,*}, \\ (f_{b_2})_{i,j}^{n+1,*} & = & \frac{1}{2} w_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*}. \end{array} \right.$$

Intersection of two outflow borders

At the intersection of two outflow borders, we propose to impose a Neumann boundary condition on $v_1 y_1 + v_2 y_2$

$$(v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} = v_1 (y_1)_{i-n_1, j-n_2}^{n+1,*} + v_2 (y_2)_{i-n_1, j-n_2}^{n+1,*}.$$

and a Dirichlet boundary condition on z_3

$$(z_3)_{i,j}^{n+1,*} = 0.$$

We need to solve the system

$$\left\{ \begin{array}{rcl} (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*} & = & v_1 \displaystyle\sum_{k=1}^{4} \lambda_k^1 (f_k)_{i,j}^{n+1,*} + v_2 \sum_{k=1}^{4} \lambda_k^2 (f_k)_{i,j}^{n+1,*} - \left({v_1}^2 + {v_2}^2\right) w_{i,j}^{n+1,*}, \\ (z_3)_{i,j}^{n+1,*} & = & \displaystyle\sum_{k=1}^{4} \left((\lambda_k^1)^2 - (\lambda_k^2)^2\right) (f_k)_{i,j}^{n+1,*}. \end{array} \right.$$

This can be written

$$\left\{ \begin{array}{rcl} v_1 \lambda_{b_1}^1 (f_{b_1})_{i,j}^{n+1,*} + v_2 \lambda_{b_2}^2 (f_{b_2})_{i,j}^{n+1,*} & = & -v_1 \lambda_{\overline{b_1}}^1 (f_{\overline{b_1}})_{i,j}^{n+1,*} - v_2 \lambda_{\overline{b_2}}^2 (f_{\overline{b_2}})_{i,j}^{n+1,*} \\ & & + \left({v_1}^2 + {v_2}^2\right) w_{i,j}^{n+1,*} + (v_1 y_1 + v_2 y_2)_{i,j}^{n+1,*}, \\ -(f_{b_1})_{i,j}^{n+1,*} + (f_{b_2})_{i,j}^{n+1,*} & = & (f_{\overline{b_1}})_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*} - \frac{(z_3)_{i,j}^{n+1,*}}{\lambda^2}. \end{array} \right.$$

The solution is given by

$$\begin{cases} (f_{b_1})_{i,j}^{n+1,*} &= \frac{1}{v_1\lambda_{b_1}^1+v_2\lambda_{b_2}^2}\Bigg(- v_1\lambda_{\overline{b_1}}^1(f_{\overline{b_1}})_{i,j}^{n+1,*} - v_2\lambda_{\overline{b_2}}^2(f_{\overline{b_2}})_{i,j}^{n+1,*} + ({v_1}^2+{v_2}^2)\, w_{i,j}^{n+1,*} \\ & \quad +(v_1y_1+v_2y_2)_{i,j}^{n+1,*} - v_2\lambda_{b_2}^2\left((f_{\overline{b_1}})_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*}\right)\Bigg), \\ (f_{b_2})_{i,j}^{n+1,*} &= \frac{1}{v_1\lambda_{b_1}^1+v_2\lambda_{b_2}^2}\Bigg(- v_1\lambda_{\overline{b_1}}^1(f_{\overline{b_1}})_{i,j}^{n+1,*} - v_2\lambda_{\overline{b_2}}^2(f_{\overline{b_2}})_{i,j}^{n+1,*} + ({v_1}^2+{v_2}^2)\, w_{i,j}^{n+1,*} \\ & \quad +(v_1y_1+v_2y_2)_{i,j}^{n+1,*} + v_1\lambda_{b_1}^1\left((f_{\overline{b_1}})_{i,j}^{n+1,*} - (f_{\overline{b_2}})_{i,j}^{n+1,*}\right)\Bigg). \end{cases}$$

Numerical tests

Order of convergence

We want to test the order of convergence of these boundary conditions. We are using the same test cases than in Section 3.2, defined in Table 3.2.

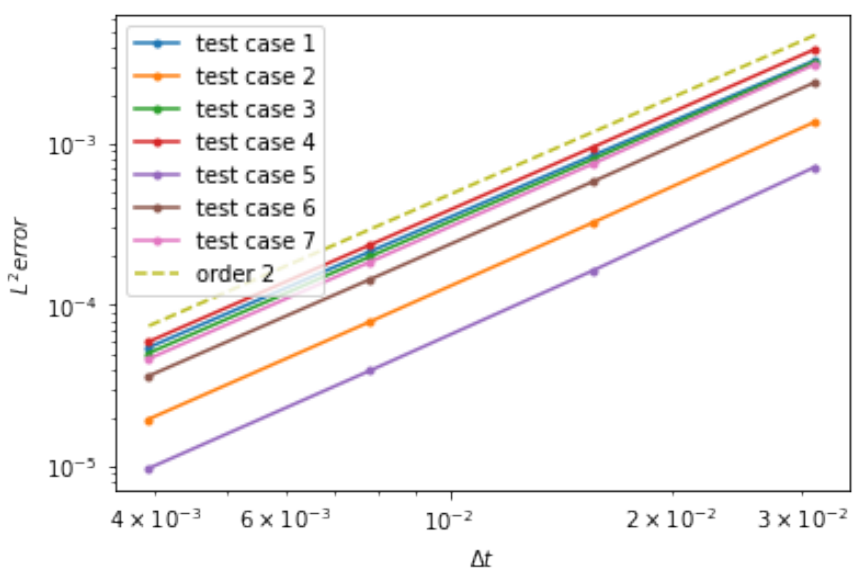

Figure 6.12: L^2 errors between the exact and the numerical solution, for the test cases defined in Table 3.2, with the second-order accurate boundary conditions.

We obtain in Figure 6.12 a convergence of order 2 for all the test cases.

Stability in a long time

Now, let us test the stability in a long time of these boundary conditions. If we compute the test case 1 with an increase of the number of time steps to $Nt = 640$, to achieve the final time $T = 20$, we obtain the evolution of the maximum of w with respect to the time in Figure 6.13. At time $T = 20$, we obtain a maximum of order 10^4. We can conclude that these boundary conditions are not stable in a long time.

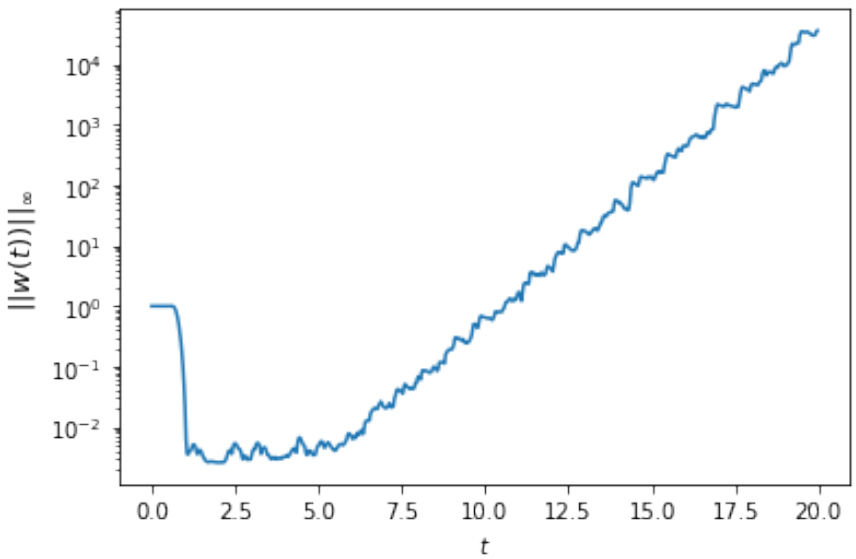

Figure 6.13: Sup norm of the solution w with respect to the time (with a log scale for the vertical axis), for the test case 1 defined in Table 3.2, with the second-order accurate boundary conditions.

Evolution of the entropy

As previously, we draw the evolution of the entropy with time, in Figure 6.14. We observe that the entropy decreases when the peak leaves the square, but then increases exponentially. These boundary conditions do not respect the decrease of the entropy, and therefore are not stable.

Projection of the second-order boundary condition on the space of the decreasing entropy boundary conditions

In Section 3.2, we have proposed boundary conditions of order 2. Unfortunately, they do not give us stability in a long time. In order to define the stable boundary conditions, we have defined a decreasing entropy condition (6.34). As we want to have boundary conditions with both properties of stability and second-order accuracy, we propose to project the second-order boundary conditions into the space of the decreasing entropy boundary conditions.

To do that, we first apply the second-order accurate boundary conditions. We denote $\tilde{\boldsymbol{f}}$, the kinetic unknown obtained. Then, for every point located on an outflow border we check that the decreasing boundary condition (6.34)

$$|\tilde{f}_b| \leqslant \sqrt{\frac{\lambda^2 + 2\boldsymbol{\lambda}_b \cdot \boldsymbol{v}}{\lambda^2 + 2\boldsymbol{\lambda}_{\bar{b}} \cdot \boldsymbol{v}}} |\tilde{f}_{\bar{b}}|,$$

is satisfied. If it is not the case, we replace $\tilde{f}_b$ by the closest value such as this decreasing entropy condition is satisfied, and we keep the other kinetic unknown. This is equivalent to

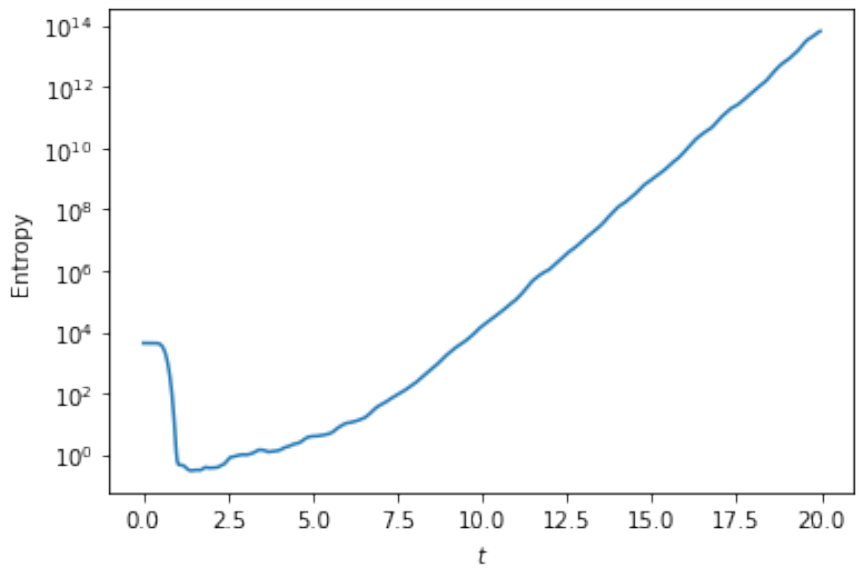

Figure 6.14: Evolution of the entropy with respect to the time (with a log scale for the vertical axis), for the test case 1 defined in Table 3.2, with the second-order accurate boundary conditions.

computing

$$\begin{cases} f_b &= \underset{|f_b| \leqslant \sqrt{\frac{\lambda^2+2\boldsymbol{\lambda}_b\cdot\boldsymbol{v}}{\lambda^2+2\boldsymbol{\lambda}_{\bar{b}}\cdot\boldsymbol{v}}}|f_{\bar{b}}|,}{\operatorname{argmin}} |f_b - \tilde{f}_b|, \\ f_i &= \tilde{f}_i, \qquad \text{for } i \neq b. \end{cases}$$

Remark 3.1. *For some of our test cases, a peak is entering into the square. It implies an increase of the entropy, coming from the inflow border. That is why we do not check the decreasing entropy condition on the inflow border. Therefore, on inflow borders, we just apply the second-order boundary condition.*

Numerical tests

Order of convergence

Let us draw in Figure 6.15 the order of convergence for the different test cases defined in Table 3.2, with this projection of the second-order boundary condition into the space of the decreasing entropy boundary conditions.

We achieve a second-order accuracy for all the test cases.

Decrease of the entropy

We apply the second-order accuracy boundary conditions only when the entropy decreasing condition is satisfied and otherwise we take a boundary condition which respects the decreasing entropy condition. Therefore, we expect the entropy to decrease at each time step. Let us draw the entropy with respect to the time of the test case 1 defined in Table 3.2.

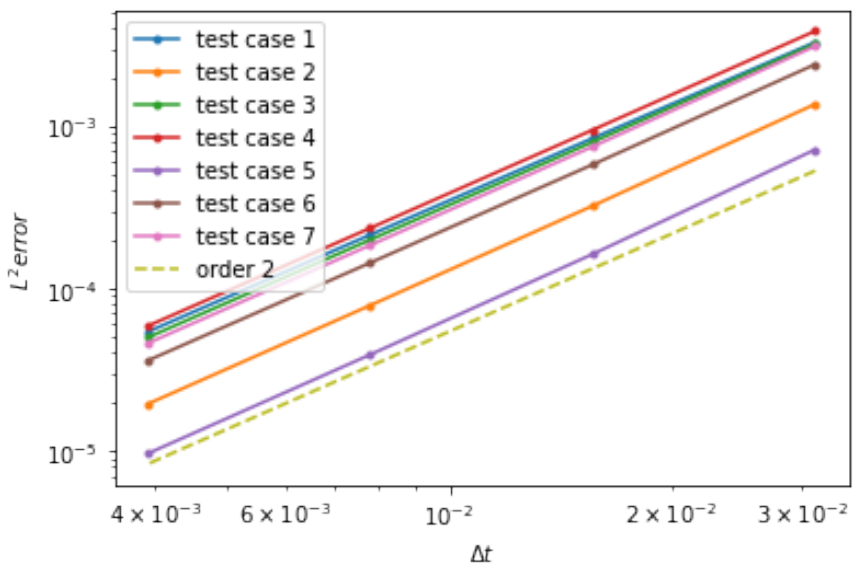

Figure 6.15: L^2 errors between the exact and the numerical solution, for the test cases defined in Table 3.2, with the projected boundary condition.

We obtain the Figure 6.16, in which we can verify the expected decrease of the entropy.

Conclusion

We can summarize the two boundary condition strategies in the following table.

Boundary conditions	Stable in a long time	Of order 2
Inflow border	Exact solution on w $z_3 = 0$	Exact solution on w
Outflow border	$\boldsymbol{\lambda}_b \cdot \boldsymbol{y} = 0$ $z_3 = 0$	Neumann on $\boldsymbol{v} \cdot \boldsymbol{y}$
Corner inflow/inflow	Exact solution on w $z_3 = 0$	Exact solution on w $z_3 = 0$
Corner inflow/outflow	Exact solution on w $\boldsymbol{\lambda}_b \cdot \boldsymbol{y} = 0$ $z_3 = 0$	Exact solution on w Neumann on $\boldsymbol{v} \cdot \boldsymbol{y}$
Corner outflow/outflow	$y_1 = 0$ $y_2 = 0$ $z_3 = 0$	Neumann on $\boldsymbol{v} \cdot \boldsymbol{y}$ $z_3 = 0$

In the first column, the decreasing entropy boundary conditions are stable but only have a first-order accuracy. The second order boundary conditions of the second column are not stable in a long time. By projecting these second-order boundary conditions into the space of the decreasing entropy boundary conditions, we obtain stability and second-order accuracy.

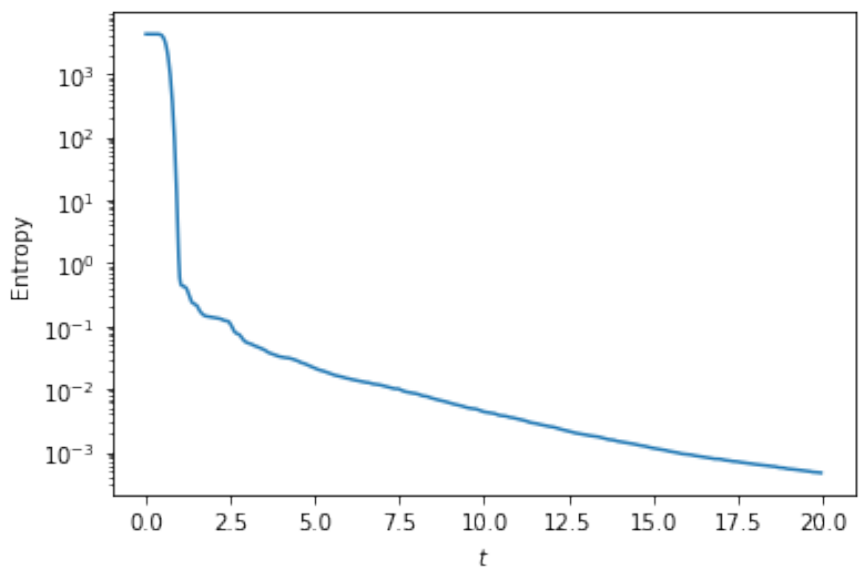

Figure 6.16: Evolution of the entropy with respect to the time (with a log scale for the vertical axis), for the test cases 1 defined in Table 3.2, with the projected boundary condition, with the projected boundary condition.

Chapter 7

Kinetic over-relaxation method for the convection equation with Fourier solver

1 Introduction

The kinetic over-relaxation method [22] is a time semi-discrete method based on the approximation of a non-linear convection equation by a set of linear transport equations with constant velocities. Very efficient, CFL-less, and accurate transport solvers like Fourier methods can be used. Moreover, the over-relaxation technic lead to second-order accuracy in time. Even higher order can be achieved by composition methods. In this paper, we apply these methods to the convection equation in two-dimension and we show that it is particularly appropriate to solve the guiding center model, where the convection velocity field is given by a solution to a Poisson equation. The guiding center model is a simplified model to describe the two-dimensional dynamics of the charge density in a Tokamak. The particles are confined in the toroidal room thanks to a large external magnetic field B. Among several dynamics, this magnetic field leads to the so-called $E \times B$ drift of the particles, where E is the self-induced electric field. This model is also equivalent to the 2d incompressible Euler equation in the vorticity formulation. The dynamics result in very fine scale structures and thus require very accurate solvers.

2 Kinetic over-relaxation approximation of the convection equation

We consider the following convection equation:

$$\partial_t \rho(t,x) + \nabla \cdot \big(\rho(t,x)\ \boldsymbol{a}(t,x)\big) = 0, \tag{7.1}$$

where $\boldsymbol{a}(t,x) \in \mathbb{R}^d$ is the velocity field and $\rho(t,x) \in \mathbb{R}$ is the convected density.

To solve this convection equation with non-constant velocity field, the relaxation method consists in approximating it with several transport equations at constant velocities. More precisely, we introduce a kinetic vector $\boldsymbol{f}(t,x) = (f_1(t,x), f_2(t,x), \ldots, f_N(t,x)) \in \mathbb{R}^N$, whose

components are associated to different velocities $(\lambda_1, \dots, \lambda_N) \in (\mathbb{R}^d)^N$. To a given kinetic vector $\boldsymbol{f}(t,x)$, we associate a macroscopic density

$$\rho_{\boldsymbol{f}}(t,x) = \sum_{i=1}^{N} f_i(t,x).$$

The numerical scheme is devised such that $\rho_{\boldsymbol{f}}$ is an approximation of the solution ρ. To this end, for any given density $\rho \in \mathbb{R}$, we introduce the so-called equilibrium kinetic vector $\boldsymbol{f}^{eq}_{[\boldsymbol{a},\rho]}$ that satisfies the following consistency relations:

$$\rho = \sum_{i=1}^{N} f^{eq}_{[\boldsymbol{a},\rho],i}, \quad \rho\boldsymbol{a} = \sum_{i=1}^{N} \lambda_i f^{eq}_{[\boldsymbol{a},\rho],i}. \tag{7.2}$$

The scheme is based on a time discretization of the following equation:

$$\partial_t \boldsymbol{f} + \sum_{k=1}^{d} \Lambda_k \partial_{x_k} \boldsymbol{f} = \frac{1}{\varepsilon}\left(\boldsymbol{f}^{eq}_{[\boldsymbol{a},\rho_{\boldsymbol{f}}]} - \boldsymbol{f}\right),$$

where $\Lambda_k = \mathrm{diag}((\lambda_1)_k, \dots, (\lambda_2)_k)$ are $N \times N$ diagonal matrices, for $k = 1, \dots, d$, and where $\varepsilon > 0$ is a small parameter that controls the distance to the equilibria set. In the time-discretization, the time-dependent relaxation operator in the r.h.s. is replaced by a projection onto the equilibria set or a symmetry with respect to the equilibria set or a combination of the two.

The time semi-discretization of the over-relaxation scheme writes as follows. We start from the equilibrium distribution associated with the initial data: $\boldsymbol{f}(0,x) = \boldsymbol{f}^{eq}_{[\boldsymbol{a}(0,x),\rho_0(x)]}$. Then, at each time step $\Delta t > 0$, starting from $\boldsymbol{f}(t,x)$, we compute $\boldsymbol{f}(t+\Delta t, x)$ in two steps:

(i) (transport step) advect the several kinetic components f_i with their respective velocities $\lambda_i \in \mathbb{R}^d$

$$f_i^*(t+\Delta t, x) = f_i(t, x - \Delta t \lambda_i), \quad \forall i \in \{1, \dots, N\},$$

which is also denoted in compact form: $\boldsymbol{f}^*(t+\Delta t, .) = T(\Delta t)\boldsymbol{f}(t,.)$.

(ii) (over-relaxation step) compute $\rho_{\boldsymbol{f}^*(t+\Delta t,.)}$ and then perform the following relaxation

$$\boldsymbol{f}(t+\Delta t, .) = \boldsymbol{f}^*(t+\Delta t, x) + \omega\left(\boldsymbol{f}^{eq}_{\left[\boldsymbol{a}(t+\Delta t,.),\rho_{\boldsymbol{f}^*}(t+\Delta t,.)\right]} - \boldsymbol{f}^*(t+\Delta t, x)\right),$$

with $\omega \in [1,2]$ a given parameter, also denoted: $\boldsymbol{f}(t+\Delta t,.) = R_\omega \boldsymbol{f}^*(t+\Delta t,.)$. For $\omega = 1$, we obtain the projection onto the equilibria set and for $\omega = 2$, we get the symmetry w.r.t the equilibria set.

The combination of these two steps writes as follows:

$$\boldsymbol{f}(t+\Delta t,.) = M_1(\Delta t)\boldsymbol{f}(t,.), \quad \text{with } M_1(\Delta t) = \Big(R_\omega \circ T(\Delta t)\Big),$$

Then $\rho_{\boldsymbol{f}}$ is a first-order approximation of the solution ρ to (7.1) for $\omega < 2$ and a second-order approximation if $\omega = 2$. We refer the reader to [22, 24] as regards the corresponding equivalent equation. From this equivalent equation, we can infer the so-called sub-characteristic condition that ensures the dissipativity of the second-order term in the expansion.

As presented in [22], higher-order time discretization can be devised by considering the following second-order time-symmetric operator:

$$M_2(\Delta t) = \left(T\left(\frac{\Delta t}{4}\right) \circ R_2 \circ T\left(\frac{\Delta t}{2}\right) \circ R_2 \circ T\left(\frac{\Delta t}{4}\right)\right),$$

and then using a palindromic composition method

$$M_p(\Delta t) = M_2(s_0\Delta t) \circ M_2(s_1\Delta t) \circ \cdots \circ M_2(s_p\Delta t),$$

where $s_i = s_{p-i}$, for $i = 0, \ldots, p$. We will consider the fourth-order Suzuki scheme ($p = 4$) and the sixth order Kahan-Li scheme ($p = 8$). We refer to [22] for the expression of the corresponding parameters.

This numerical scheme has the advantage to concentrate all the non-linear operators in a local step, while the transport step becomes fully linear. Therefore, CFL-less method can be employed to make these transport steps. A semi-Lagrangian scheme has been used in [24]. On non-Cartesian meshes, implicit Discontinuous Galerkin method with upwind fluxes can be used as proposed in [22]. Here, we consider a Fourier discretization of the transport equation, ensuring a spectral accuracy.

In the sequel, we will use the so-called [D2Q4] kinetic approximation ($N = 4$). It consists in introducing the four velocities directed along the Cartesian axes:

$$\lambda_1 = \begin{bmatrix}\lambda\\0\end{bmatrix}, \quad \lambda_2 = \begin{bmatrix}0\\\lambda\end{bmatrix}, \quad \lambda_3 = \begin{bmatrix}-\lambda\\0\end{bmatrix}, \quad \lambda_4 = \begin{bmatrix}0\\-\lambda\end{bmatrix},$$

with $\lambda > 0$ and then we define the kinetic equilibrium vector:

$$f^{eq}_{[\boldsymbol{a},\rho],i} = \frac{\rho}{4} + \frac{\rho(\boldsymbol{a}\cdot\lambda_i)}{2\lambda^2}, \quad \forall i \in \{1,2,3,4\}.$$

This is the only solution to consistency relations (7.2), which satisfies symmetries. The sub-characteristic condition writes in that case: $\lambda > \max_{[0,T]\times\Omega} ||\boldsymbol{a}(t,\boldsymbol{x})||$, where $[0,T]\times\Omega$ is the computational domain.

We will also consider the [D2Q5] kinetic approximation ($N = 5$), where a fifth central null velocity is added:

$$\lambda_1 = \begin{bmatrix}\lambda\\0\end{bmatrix}, \quad \lambda_2 = \begin{bmatrix}0\\\lambda\end{bmatrix}, \quad \lambda_3 = \begin{bmatrix}-\lambda\\0\end{bmatrix}, \quad \lambda_4 = \begin{bmatrix}0\\-\lambda\end{bmatrix}, \quad \lambda_5 = \begin{bmatrix}0\\0\end{bmatrix}.$$

where $\lambda > 0$. The kinetic equilibrium vector has to satisfy consistency relations (7.2):

$$\rho = f^{eq}_{[\boldsymbol{a},\rho],1} + f^{eq}_{[\boldsymbol{a},\rho],2} + f^{eq}_{[\boldsymbol{a},\rho],3} + f^{eq}_{[\boldsymbol{a},\rho],4} + f^{eq}_{[\boldsymbol{a},\rho],5},$$
$$\rho a_1 = \lambda(f^{eq}_{[\boldsymbol{a},\rho],1} - f^{eq}_{[\boldsymbol{a},\rho],3}), \quad \rho a_2 = \lambda(f^{eq}_{[\boldsymbol{a},\rho],2} - f^{eq}_{[\boldsymbol{a},\rho],4}).$$

This system is underdetermined. As already proposed in [24] for the one-dimensional case, we consider the following decomposition based on a flux-splitting

$$f^{eq}_{[\boldsymbol{a},\rho],i} = \rho\,(\lambda_i\cdot\boldsymbol{a})_+\,, \quad \forall i \in \{1,2,3,4\}, \quad f^{eq}_{[\boldsymbol{a},\rho],5} = \rho - \sum_{i=1}^{4} f^{eq}_{[\boldsymbol{a},\rho],i}\,,$$

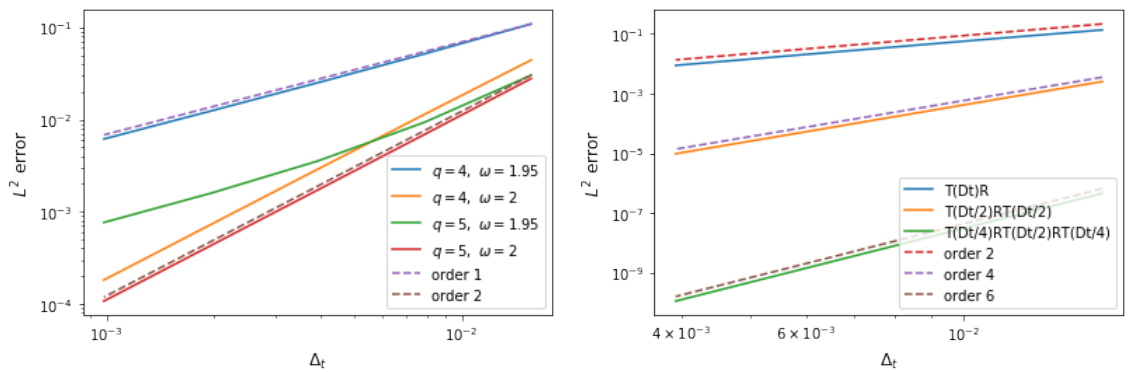

Figure 7.1: L^2 error between the exact and the numerical solution obtained as function of the time step. Left: Comparison between $q = 4$ ([D2Q4]) and $q = 5$ ([$D2Q5$], $r = 4$) for different ω. Right: Comparison between different splitting operators when using the [$D2Q5$] method ($r = 4$) with a Kahan-Li palindromic composition and $\omega = 2$. Parameters: $\lambda = 2.1$, $N_x = N_y = 200$.

where for any $v \in \mathbb{R}$, $v_+ = \max\{v, 0\}$ stands for the positive part of v or can be approximated by a smooth version $v_+ = (v+H_r(v))/2$ where $H_r(v)$ are Halley's functions defined recursively by: $H_0(x) = 1$, $H_{r+1}(x) = H_r(x)(H_r(x)^2 + 3x^2)/(3H_r(x)^2 + x^2)$. The sub-characteristic condition is the same as for the [D2Q4] approximation. As explained in [24], this scheme is expected to be more precise and better captures unidirectional flows.

3 Numerical results

In this section, we validate the numerical scheme on two test-cases : the rotation advection test-case and the Kelvin-Helmholtz test-case for the guiding-center model. In these two test-cases, the transport part $T(\Delta t)$ is discretized with a Fourier method.

Rotation test-case We consider the convection equation (7.1) where the velocity field is given by $\boldsymbol{a}(\boldsymbol{x}) = \boldsymbol{x}^\perp$. This velocity field is divergence free: $\nabla \cdot \boldsymbol{a} = 0$. Therefore, the convection equation (7.1) is equivalent to the advection equation:

$$\partial_t \rho(t, \boldsymbol{x}) + \boldsymbol{a}(\boldsymbol{x}) \cdot \nabla \rho(t, \boldsymbol{x}) = 0,$$

and the exact solution is just the rotation of the initial density around the origin.

In the following, we consider the domain $\Omega = [-1, 1] \times [-1, 1]$ and the exact solution:

$$\rho(t, \boldsymbol{x}) = \frac{1}{2\pi} \exp\left(-\frac{\|R(t)(\boldsymbol{x} - \boldsymbol{x}_0)\|^2}{\sigma^2}\right).$$

where $\sigma = 0.1$ and $\boldsymbol{x}_0 = (0.5, 0)$ and $R(t)$ is the rotation matrix of angle t. We use $N_x = N_y = 200$ discretization points in each direction.

Figure 7.1 (left) shows that the [D2Q4] scheme is first order accurate with the relaxation parameter $\omega = 1.95$. Second order accuracy is achieved with $\omega = 2$ for both [D2Q4] and

Table 7.1: Number of time steps and execution times needed to achieve an accuracy of 10^{-8} at time $T = \pi/2$ with $\lambda = 2.1$, $N_x = N_y = 200$.

	Nb of time steps	Nb of transport steps	Error L^2	Execution time
$[D2Q5]$, $\omega = 2$, M_1	172000	172000	9.258×10^{-9}	4985.282
$[D2Q5]$, $\omega = 2$, M_2	82000	246000	9.975×10^{-9}	6298.363
$[D2Q5]$, $\omega = 2$, M_2, Suzuki	570	8 550	9.516×10^{-9}	223.407
$[D2Q5]$, $\omega = 2$, M_2, Kahan-Li	190	5130	9.924×10^{-9}	145.426
$[D2Q4]$, $\omega = 2$, M_2, Kahan-Li	215	5 805	9.627×10^{-9}	132.539

[D2Q5] using the M_1 operator. As expected, we also note that the [D2Q5] is more accurate than the [D2Q4] scheme.

In Figure 7.1 (right), we observe that the M_2 operator is required to obtain the sixth-order accuracy of the Kahan-Li composition method. Using the M_1 operator leads to a second order operator and the Strang splitting $M_2^S(\Delta t) = \left(T\left(\frac{\Delta t}{2}\right) \circ R_2 \circ T\left(\frac{\Delta t}{2}\right)\right)$ to a fourth-order accuracy only.

As regards the computational time, we observe in Table 7.1 that considering M_1 is 1.26 times more efficient than considering M_2 when $\omega = 2$. Indeed, both methods are of order 2 and M_1 requires less transport steps. However, using M_2, we can use the Suzuki or the Kahan-Li composition methods that are respectively 22 and 34 times faster. For these comparisons, we use the $[D2Q5]$ method. The $[D2Q4]$ method seems just as fast even though it requires more transport steps. Although more accurate, the [D2Q5] is slowed down by the evaluation of the Halley functions.

Kelvin-Helmholtz test-case We consider the guiding center model that describes the two-dimensional dynamics of electrons resulting from the $E \times B$ drift due to a large magnetic field. Their charge density is denoted $\rho(t,x) > 0$ and the guiding center model writes :

$$\partial_t \rho + E^\perp \cdot \nabla \rho = 0, \tag{7.3}$$

$$-\Delta \phi = \rho_0 - \rho, \qquad E = -\nabla \phi. \tag{7.4}$$

where $E(t,x) \in \mathbb{R}^d$ the electric field and $\phi(t,x) \in \mathbb{R}$ the electric potential. $\rho_0(t) > 0$ denotes the ion background charge density, which is supposed homogeneous. Actually, this model is equivalent to the 2d incompressible Euler equation in the vorticity formulation. Here we consider a square domain Ω with periodic boundary conditions and we thus assume that $\rho_0(t)$ equals the average of the density over the domain: $\rho_0(t) = \frac{1}{|\Omega|}\int_\Omega \rho(t,x)dx$.

Since $E = -\nabla\phi$, the advection vector field $E^\perp$ is divergence free. The transport equation is thus equivalent to the conservative convection equation

$$\partial_t \rho + \nabla \cdot \left(\rho E^\perp\right) = 0, \tag{7.5}$$

Unlike the previous advection equations presented so far, here the advection field depends on the density itself. Therefore, the over-relaxation scheme is slightly modified and writes:

(i) (transport step) $f^*(t+\Delta t,.) = T(\Delta t)\boldsymbol{f}(t,.)$,

(ii) (Poisson step) compute $\rho_{\boldsymbol{f}^*(t+\Delta t},.)$ and then find $\phi^*(t+\Delta t,.)$ by solving the Poisson equation and then $\boldsymbol{a}(t+\Delta t,.) = E^*(t+\Delta t,.)^\perp$.

(iii) (over-relaxation step) $f(t+\Delta t,.) = R_\omega f^*(t+\Delta t,.)$.

Note that both the transport step and the Poisson equation can be solved using a Fourier discretization in the square domain.

As already considered in [82, 85, 27], the Kelvin-Helmholtz instability test-case consists in considering the following initial condition:

$$\rho_{init}(x,y) = \sin x + \varepsilon\, \cos(ky),$$

in the domain $[0,2\pi]\times[0,2\pi/k]$, with periodic boundary conditions, and where $k\in\mathbb{R}$ is the perturbation wave number and $\varepsilon>0$ is the perturbation amplitude. This is a perturbation of the stationary solution $\rho_0(x)=\sin x$ and $\phi_0(x) = -\sin x$. According to [82], there exists a critical wave number $k_s = 1$ such that an instability develops only for $k<k_s$. The instability rates are not known explicitly. However, we can compute them numerically.

We look for solutions of the form

$$\rho(x,y,t) = \rho_0(x) + \varepsilon\rho_1(x,y,t), \quad \phi_0(x,t) = \phi_0(x) + \varepsilon\, \phi_1(x,y,t)$$

where $\rho_1(x,y,t) = \tilde{\rho}_1(x)\exp(iky)\exp(-i\omega t)$, $\phi_1(x,y,t) = \tilde{\phi}_1(x)\exp(iky)\exp(-i\omega t)$. Following [82], it can be proved that $\tilde{\phi}_1$ solves the generalized eigenvalue problem:

$$\phi_0'\left(\partial_{x^2}\tilde{\phi}_1 - k^2\tilde{\phi}_1\right) + \tilde{\phi}_1\rho_0' = -\omega/k\,\phi_0'\left(\partial_{x^2}\tilde{\phi}_1 - k^2\tilde{\phi}_1\right), \tag{7.6}$$

in which ω/k stands for the eigenvalue. As explained in [82], it can be proved that unstable solutions, corresponding to ω/k with positive imaginary part, exist if and only if $k<k_s=1$. For k near k_s, a first order approximation of the instability rate can be computed: $\omega/k = 2(k_s-k)i$. Alternatively, we can also compute the instability rate by solving (7.6) numerically using a finite difference method. Introducing a space step $\Delta x = 1/N$ with $N\in\mathbb{N}$ and the corresponding spatial discretization of the interval $[0,1]$, $x_i = i\Delta x$, we consider the approximate solution $\Phi_1\in\mathbb{C}^N$, such that $(\Phi_1)_i \approx \tilde{\phi}_1(x_i)$ and which solves the following problem

$$C\left(D + (1-k^2)\mathrm{Id}\right)\Phi_1 = \omega/k\left(D - k^2\mathrm{Id}\right)\Phi_1, \tag{7.7}$$

where $C = \mathrm{diag}(\cos(x_1),\dots,\cos(x_N))$ is diagonal matrix and D is discrete Laplacian matrix with periodic boundary conditions. Therefore, assembling $A = C\,(D+\ (1-k^2)\mathrm{Id})$ and $B = (D-k^2\mathrm{Id})$, we just have to compute numerically the eigenvalues of the matrix $B^{-1}A$ and then keep the one with the largest imaginary part.

In Figure 7.2 (left) is plotted the time evolution of the k-th Fourier mode of the potential. The instability rate fits perfectly with the expected one obtained solving (7.7). In the middle and right are plotted the contour lines of the density with the first-order scheme M_1 and the Kahan-Li composition methods. This illustrates the need to use high order scheme to capture the small structures.

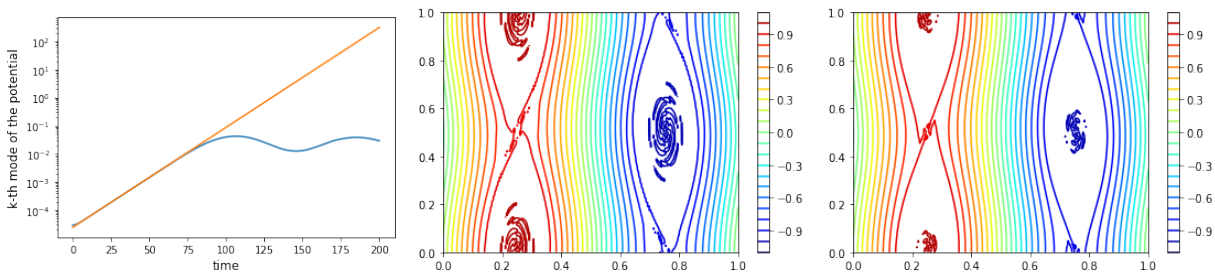

Figure 7.2: (Kelvin-Helmholtz, $k = 0.95$, $\varepsilon = 10^{-4}$, $N_x = N_y = 200$, $\Delta t = 0.01$, [D2Q5], $r = 4$, $\lambda = 2.02$) Left: Time evolution of the k-th Fourier mode of the potential (in blue) and the straight line with slope $\mathrm{Im}(\omega) = 0.08185$ (in orange) with Kahan-Li, $\omega = 2$. Middle and left: Contour lines of the density at final time $T = 200$ with $\omega = 2$, Kahan-Li (middle) and $\omega = 1.95$, M_2 (right).

4 Conclusion

In this paper, we show that the kinetic over-relaxation method enables to devise numerical schemes for the convection equation based on Fourier discretization. The proposed method is optimally high-order accurate in space and can reach sixth order time accuracy with the Kahan-Li composition method. Unless high order schemes require more intermediate transport steps, the computational cost can be drastically decreased. Moreover, the method has been extended to the non-linear guiding center model. This is the first step before the extension to more complex advection equations like the gyro-kinetic equation in plasma physics.

Chapter 8

Parallel kinetic scheme for transport equations in complex toroidal geometry

1 Introduction

Physics of tokamak raise a lot of computational challenges. Indeed, a tokamak is an experimental device designed for creating hot charged particle gas: in such a device, the charged particles are confined thanks to a large magnetic field. Their dynamics involve several multi-scale transport phenomena, which introduces stringent constraints on the discretization parameters and requires high-order schemes. Moreover, tokamak generally has a toroidal shape and presents a cylindrical symmetry around the vertical axis. The geometry of the poloidal plane, however, can be complicated and unstructured meshes are required. We thus aim at proposing an efficient numerical scheme for solving conservative transport equations in such three-dimensional toroidal geometries.

In tokamak, the large toroidal magnetic field results in a scale separation between toroidal and poloidal dynamics. Indeed, particles exhibit fast transport in the toroidal direction, while the poloidal dynamics follow incompressible dynamics. Explicit numerical scheme would impose the dependence of the time step to the mesh through the Courant-Friedrichs-Lévy (CFL) stability condition. Thus, to avoid too small time steps, the mesh should be perfectly adapted to the very complex three-dimensional dynamics. This is complicated in practice because the creation of the mesh is often an independent step of the simulation.

To dissociate the issues of numerical parameters from the mesh construction, CFL-less numerical methods have been proposed. For transport equations, one of the main such methods is the semi-Lagrangian one, introduced in [85]. Each iteration consists in computing the foot of the characteristics issued from the mesh nodes and then interpolating the solution at these points. Several variants have been proposed, see for instance [11, 62]. This method has been successfully used for full tokamak simulations in the Gysela code [51]. In these simulations, the computational domain is a torus with circular sections discretized with a polar mesh. Unfortunately, the standard semi-Lagrangian method does not easily handle full unstructured meshes, since stability and conservation issues arise. It is therefore difficult to extend the method to more general geometries.

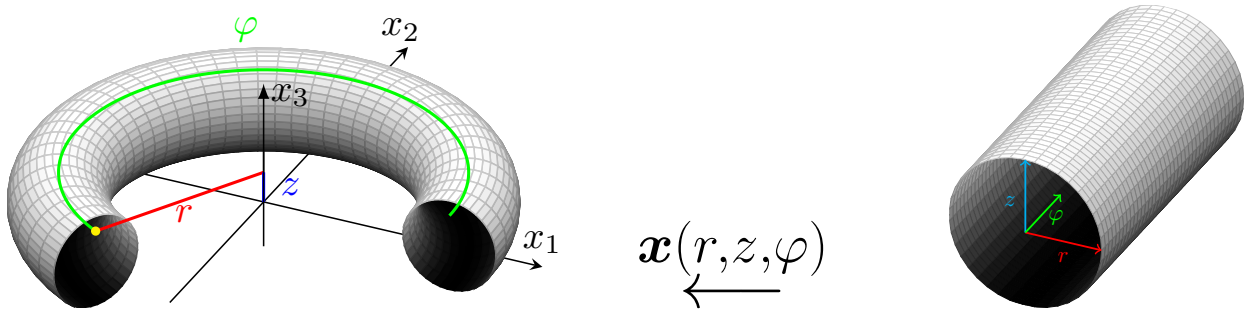

Figure 8.1: Cartesian coordinates (x_1, x_2, x_3) and cylindrical coordinates (r, z, φ).

Indeed, the shapes of the tokamak lead naturally to specific meshes. The domain being axisymmetric, the mesh can be structured in the toroidal direction. However, the poloidal sections of tokamak can be either circular like in the Tore Supra device [87] but can also have very complex geometry. The poloidal plane of the tokamak ITER has a so-called D-shape. This geometry is in fact chosen to optimize the confinement. For instance, the poloidal mesh can be constructed as a multi-patch Bézier mesh [52].

The numerical methods must naturally be adapted to such meshes both in their structure and parallelism. After rewriting the conservative transport equation in the cylindrical coordinates (r, z, φ) (see Figure 8.1), very simple numerical methods can be envisaged in the φ direction (e.g. Fourier or semi-Lagrangian method). But more sophisticated solvers have to be considered to deal with the transport in unstructured meshes in (r, z) planes.

In this work, we propose to adapt the CFL-less kinetic solver proposed in [22] to the toroidal geometry. The method reuses ideas coming from the Lattice-Boltzmann Method (LBM) [79, 20, 97, 38] or from the kinetic schemes [13, 6]. LBM have been introduced to solve the incompressible Navier-Stokes equation as well as advection-diffusion equations while kinetic schemes were introduced to solve hyperbolic systems. Note that generalized LBM have also been proposed for kinetic equations [31, 32] . The main point is to replace the conservative transport equation, where the velocity field is not constant, by a **few transport equations at constant velocities**, coupled by a stiff local source term. The coupled system is then solved with a splitting algorithm that separates the free transport steps and the stiff source terms. The free transport steps are easier to solve, because the transport step is done at constant velocity.

Here, we consider six constant velocities aligned to the Cartesian axes in the cylindrical coordinates, whose magnitudes may be different in the poloidal and toroidal directions: this is a so-called D3Q6 model. The magnitudes have to satisfy a so-called sub-characteristic condition to ensure the stability of the numerical scheme. We would like to emphasize that this condition does not involve the time and space discretization parameters. Taking different magnitudes enables us to handle a different scale between transport in poloidal and toroidal directions. Other sets of velocities have been proposed to better capture multi-scale effects [24, 37] but stability is then more delicate to study. We also refer to [36] for recent analysis of such schemes.

In order to make the time step independent from the possibly complicated poloidal mesh, the transport equations at constant velocities in poloidal planes are solved using a CFL-less

implicit DG method [22]. The implicit method has actually an explicit cost because the transport velocities are constant: the transport scheme is just to invert a block-triangular linear system. For the transport in the toroidal directions, since the mesh is uniform, an exact transport solve is chosen like in Lattice-Boltzmann methods. These choices link the time step to the toroidal space discretization but still keep it independent from the poloidal mesh.

The whole method is conservative. Moreover, it is possible, using an adequate splitting algorithm with a so-called over-relaxation techniques [28] to achieve second-order accuracy in time. For Lattice Boltzmann schemes, this has been analyzed in [33]. High order spatial discretization is considered in the poloidal planes thanks to the DG method, while transport is exact in the toroidal direction. Thus, we finally obtain a second-order scheme well adapted to manage different dynamics in the poloidal and toroidal directions.

The resulting scheme has also nice parallelization possibilities. In a poloidal plane, the block-triangular linear systems resulting from the DG scheme that are well solved by an optimized task-based implementation [7, 18]. In the toroidal direction, the transport equations are solved by a simple shift operator. It is here implemented by simple MPI point-to-point communications.

In the following, we first present the whole mathematical, numerical and programming construction. We then verify its accuracy and efficiency with some pure transport test cases. Finally, in order to assess the usefulness of the method in more complex framework, we apply our transport solver to the numerical simulation of the diocotron instability in two-dimensional and three-dimensional configurations. This plasma physics test cases require to couple the transport solver with a Poisson solver and make the whole model non-linear.

2 Transport equation and reformulation

We consider the following conservative transport equation:

$$\partial_t \sigma + \nabla_{\boldsymbol{x}} \cdot (\sigma \boldsymbol{v}) = 0, \tag{8.1}$$

where the unknown $\sigma(\boldsymbol{x}, t)$ depends on a space variable $\boldsymbol{x} \in \mathbb{R}^3$ and of the time t. The velocity field $\boldsymbol{v}(\boldsymbol{x}, t)$ is given. As we are interested in solving this equation in a toroidal domain, we first rewrite an equivalent formulation in cylindrical coordinates. We then present its approximation by a system of transport equation at constant velocities using the Lattice-Boltzmann Method.

2.1 In cylindrical coordinates

Let us first consider an arbitrary change of coordinates $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{r})$. The Jacobian of this change of variables is denoted by $j(\boldsymbol{r}) = \det \boldsymbol{x}'(\boldsymbol{r})$. In this new set of coordinates, the transport equation (8.1) becomes:

$$\partial_t \rho + \nabla_{\boldsymbol{r}} \cdot (\rho \boldsymbol{u}) = 0. \tag{8.2}$$

where the new unknown $\rho(\boldsymbol{x},t)$ and the new velocity field $\boldsymbol{u}(\boldsymbol{x},t)$ are defined by:

$$\rho(\boldsymbol{r},t) = j(\boldsymbol{r})\sigma(\boldsymbol{x}(\boldsymbol{r}),t), \tag{8.3}$$
$$\boldsymbol{u}(\boldsymbol{r},t) = \boldsymbol{x}'(\boldsymbol{r})^{-1}\boldsymbol{v}(\boldsymbol{x}(\boldsymbol{r}),t). \tag{8.4}$$

For tokamak applications, we are particularly interested in the change from Cartesian to cylindrical coordinates. We denote by $\boldsymbol{x} = (x_1, x_2, x_3)^T$ the Cartesian coordinates and by $\boldsymbol{r} = (r, z, \varphi)^T$ the cylindrical coordinates. The change of variables is given by

$$\begin{aligned} x_1 &= r\cos\varphi, \\ x_2 &= r\sin\varphi, \\ x_3 &= z. \end{aligned}$$

We also define the cylindrical frame

$$\boldsymbol{e}_r = \begin{pmatrix} \cos\varphi \\ \sin\varphi \\ 0 \end{pmatrix}, \quad \boldsymbol{e}_z = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad \boldsymbol{e}_\varphi = r\begin{pmatrix} -\sin\varphi \\ \cos\varphi \\ 0 \end{pmatrix}.$$

which is the columns of $\boldsymbol{x}'(\boldsymbol{r})$. Note that the vector $\boldsymbol{e}_\varphi$ is not a unit vector: its Euclidean norm is equal to r. From (8.3), we define the new unknown:

$$\rho(\boldsymbol{r},t) = r\sigma(\boldsymbol{x},t).$$

Then we introduce the velocity fields $\boldsymbol{u} = (u_r, u_z, u_\varphi)^T$ as defined by (8.4). It is related to $\boldsymbol{v}$ by the following relation:

$$\boldsymbol{v}(\boldsymbol{x},t) = \begin{pmatrix} v_1(\boldsymbol{x},t) \\ v_2(\boldsymbol{x},t) \\ v_3(\boldsymbol{x},t) \end{pmatrix} = u_r(\boldsymbol{r},t)\,\boldsymbol{e}_r + u_z(\boldsymbol{r},t)\,\boldsymbol{e}_z + u_\varphi(\boldsymbol{r},t)\,\boldsymbol{e}_\varphi.$$

Up to a change of unknowns, the initial conservative transport equation (8.1) has an equivalent conservative formulation (8.2) in cylindrical coordinates. Therefore, it is possible to solve the transport equation in cylindrical coordinates as we were in Cartesian coordinates. In particular equation (8.2) can be rewritten

$$\partial_t\rho + \boldsymbol{\nabla}_{(r,z)} \cdot \left(\rho\begin{pmatrix} u_r \\ u_z \end{pmatrix}\right) + \partial_\varphi(\rho u_\varphi) = 0, \tag{8.5}$$

and as proposed in this work, we can use a different discretization for the (r, z) and the φ coordinates. Indeed, the geometry will be supposed to be invariant in the φ direction, which will allow several optimizations.

2.2 Kinetic relaxation method

For solving the transport equation (8.2), the kinetic relaxation method consists in introducing n_v unknowns $f_k(\boldsymbol{r},t)$ associated to n_v constant kinetic velocities in cylindrical coordinates $\boldsymbol{\lambda}_k$, $k = 0 \dots n_v - 1$. The $n_v - 2$ first velocities are used for solving the transport problem

in the (r, z) plane and the last two velocities are used for the φ direction. For instance, we can take $n_v = 6$, with 4 velocities in the poloidal plane

$$\boldsymbol{\lambda}_0 = \begin{pmatrix} \lambda_p \\ 0 \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_1 = \begin{pmatrix} -\lambda_p \\ 0 \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_2 = \begin{pmatrix} 0 \\ \lambda_p \\ 0 \end{pmatrix}, \quad \boldsymbol{\lambda}_3 = \begin{pmatrix} 0 \\ -\lambda_p \\ 0 \end{pmatrix},$$

and 2 velocities in the toroidal direction

$$\boldsymbol{\lambda}_4 = \begin{pmatrix} 0 \\ 0 \\ \lambda_t \end{pmatrix}, \quad \boldsymbol{\lambda}_5 = \begin{pmatrix} 0 \\ 0 \\ -\lambda_t \end{pmatrix},$$

where $\lambda_p > 0$ denotes the norm of the kinetic velocities of the poloidal plane and $\lambda_t > 0$ the norm of the velocities in the toroidal direction. This choice corresponds to the so-called D3Q6 method. We could also consider only 3 velocities in the poloidal planes which would lead to a D2Q3 method in the poloidal plane coupled to a D1Q2 method in the toroidal direction. Then the kinetic relaxation method consists in considering the following kinetic model

$$\partial_t f_k + \boldsymbol{\lambda}_k \cdot \boldsymbol{\nabla}_{\boldsymbol{r}} f_k = \frac{1}{\tau}(f_k^{eq} - f_k), \tag{8.6}$$

with $\tau > 0$ is the relaxation parameter and f_k^{eq} is the so-called equilibrium kinetic distribution defined by

$$f_k^{eq} = \frac{\rho}{n_v} + \frac{\rho \boldsymbol{u} \cdot \boldsymbol{\lambda}_k}{2\|\boldsymbol{\lambda}_k\|^2}, \tag{8.7}$$

with $\boldsymbol{u}$ is the velocity field defined in (8.4) but here ρ is defined by

$$\rho = \sum_{k=0}^{n_v-1} f_k. \tag{8.8}$$

The equilibrium kinetic distribution is chosen such that equation (8.6) is an approximation of equation (8.2). More precisely, in the limit where τ goes to 0, the density ρ defined by (8.8) tends to the solution to (8.2). This approximation is stable under a so-called sub-characteristic condition. Roughly speaking, it states that the ellipsoid, whose main axes are aligned to the kinetic speeds $\boldsymbol{\lambda}_k$, has to contain the velocity field $\boldsymbol{u}$. More precisely, the model is stable if all the velocities $\boldsymbol{u} = (u_r, u_z, u_\varphi)^T$ satisfy the following condition

$$\frac{u_r^2}{\lambda_p^2} + \frac{u_z^2}{\lambda_p^2} + \frac{u_\varphi^2}{\lambda_t^2} \leqslant \frac{1}{3}.$$

The computations are given in the appendix 6. For more details, we refer to [6, 7, 13, 22, 33].

The main interest to transform (8.5) into (8.6) is that now the transport velocities are **constant**. This will dramatically simplify the resolution of the equations and allows designing explicit schemes that are unconditionally stable.

3 Transport solver

In this section we focus on the numerical schemes used for solving the transport equations involved in the kinetic model (8.6):

$$\partial_t f_k + \boldsymbol{\lambda}_k \cdot \boldsymbol{\nabla}_{\boldsymbol{r}} f_k = 0, \tag{8.9}$$

We want to solve them in domains with a cylindrical geometry. Indeed, we recall that, up to a change of variable (see section (2.1)), it enables us to consider toroidal domains as required for tokamak simulations. The starting point is thus a two-dimensional unstructured mesh of one poloidal plane ($\varphi = \text{Cst}$). Here it is chosen made of second order curved quadrilaterals with eight nodes ("Q8" family in the finite element terminology). Then this mesh is extruded in the φ direction leading to a uniform structured mesh of space step $\Delta\varphi$ in that direction.

Two different transport solvers are used. For transport in the toroidal direction, we consider an exact solver to take advantage of the uniform mesh: this corresponds to the Lattice-Boltzmann method (LBM). For transport in the poloidal planes, we consider an implicit DG scheme. We first details the two numerical schemes and then briefly present their parallelization.

3.1 Transport in the toroidal direction: LBM

Only the last two kinetic components, f_{n_v-2} and f_{n_v-1}, are transported in the toroidal direction. We solve these transport equations with a simple shift, because of the structured mesh. This writes:

$$(f_{n_v-2})_j^{n+1} = (f_{n_v-2})_{j-1}^n, \quad (f_{n_v-1})_j^{n+1} = (f_{n_v-1})_{j+1}^n,$$

where j refers to the poloidal planes index. This requires to link the time step to the space step in the φ direction: $\Delta t = \Delta\varphi/\lambda_t$. This is the classical algorithm used in the Lattice-Boltzmann method.

3.2 Transport in the poloidal direction: implicit DG

Because the poloidal shape can be arbitrary, we have introduced an unstructured mesh and thus we consider a Discontinuous Galerkin (DG) approximation for transporting the $n_v - 2$ first kinetic components $f_0, \ldots, f_{n_v-3}$ whose associated kinetic velocities are poloidal. In order to avoid stability constraints due to small mesh elements, we consider an implicit solver.

Implicit DG formulation

The objective is to solve a two-dimensional transport equation, with constant velocity

$$\partial_t f + \boldsymbol{\lambda} \cdot \boldsymbol{\nabla}_{(r,z)} f = 0.$$

where f and $\boldsymbol{\lambda}$ refer to one of the $n_v - 2$ first kinetic component and its associated kinetic velocity. In each cell L, we consider polynomial basis functions ψ_i^L of degree p. For efficiency

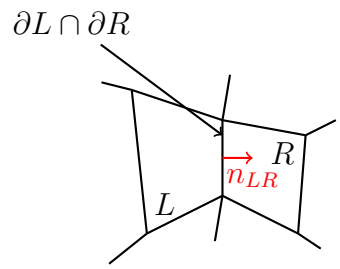

Figure 8.2: Normal vector convention.

reason, the basis functions are Lagrange polynomials based on Gauss-Lobatto quadrature points. The transported function f is approximated in cell L by a linear expansion on the basis functions

$$f(r,z,n\Delta t) \simeq f_L^n(r,z) = \sum_j f_{L,j}^n \psi_j^L(r,z), \quad (r,z) \in L.$$

The unknowns are the coefficients $f_{L,j}^n$ of the linear expansion.

In order to avoid constraining CFL conditions, we only envisage implicit solvers. We first consider the simplest first order implicit DG approximation scheme. It reads: $\forall L, \forall i$,

$$\int_L \frac{f_L^n - f_L^{n-1}}{\Delta t}\psi_i^L - \int_L \boldsymbol{\lambda}\cdot\boldsymbol{\nabla}\psi_i^L f_L^n + \int_{\partial L} \left((\boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR})^+ f_L^n + (\boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR})^- f_R^n\right)\psi_i^L = 0. \tag{8.10}$$

where the basis functions ψ_i^L play the role of test functions, R denotes the neighbor cells along ∂L, $\boldsymbol{n}_{LR}$ is the unit normal vector on ∂L oriented from L to R (see Figure 8.2) and $(\boldsymbol{\lambda}\cdot\boldsymbol{n})^+ = \max(\boldsymbol{\lambda}\cdot\boldsymbol{n}, 0)$, $(\boldsymbol{\lambda}\cdot\boldsymbol{n})^- = \min(\boldsymbol{\lambda}\cdot\boldsymbol{n}, 0)$. We thus use an upwind numerical flux. For more details on the DG method, we refer for instance to [56, 7]. We can rewrite (8.10) in the matrix form

$$M_L f_L^n = M_L f_L^{n-1} + \Delta t \left(A_L f_L^n - \sum_{\substack{LR\in\partial L,\\ \boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR}>0}} B_{LR} f_L^n - \sum_{\substack{LR\in\partial L,\\ \boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR}<0}} C_{LR} f_R^n \right), \tag{8.11}$$

where M_L denotes the mass matrix, A_L the volume advection matrix, B_{LR} and C_{LR} the flux advection matrices between cell L and downwind or upwind cells R: cell R is said to be upwind with respect to cell L if $\boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR} < 0$ and downwind otherwise. Their coefficients are given by

$$(M_L)_{i,j} = \int_L \psi_i^L \psi_j^L, \quad (A_L)_{i,j} = \int_L (\boldsymbol{\lambda}\cdot\nabla\psi_i^L)\,\psi_j^L,$$

$$(B_{LR})_{i,j} = \int_{\partial L} (\boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR})\psi_i^L\psi_j^L \quad \text{and} \quad (C_{LR})_{i,j} = \int_{LR} (\boldsymbol{\lambda}\cdot\boldsymbol{n}_{LR})\psi_i^L\psi_j^R.$$

In the above formalism, we only describe a first order time scheme. In practice, we actually use a second order implicit Crank-Nicolson time stepping. But it is very similar to the above description.

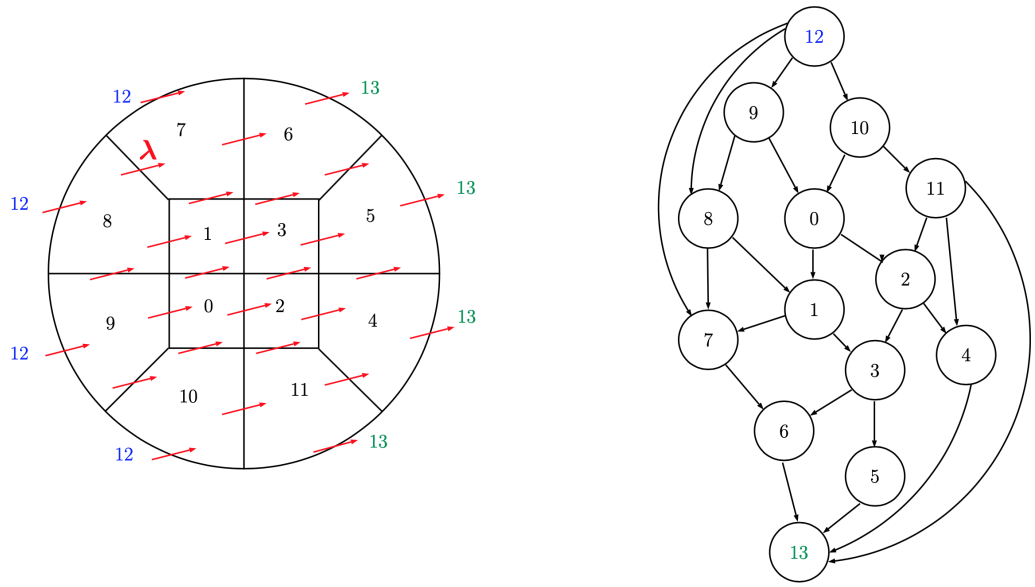

Figure 8.3: Left: Unstructured mesh of the disk, with a constant velocity field $\boldsymbol{\lambda}$ (in red). The velocity field is chosen to have a slight angle with respect to the horizontal axis to better highlight the dependency between cells. Label 12 represent where upwind boundary conditions should be provided. Right: Associated dependency graph. Transport is performed from top to bottom nodes.

Downwind algorithm

One time-step of the implicit DG scheme consists in computing the distribution function at time t^n, f^n from the distribution function f^{n-1} of the previous time step. From (8.10) it is clear that one has to solve a linear system, as in any implicit scheme. However, because the kinetic velocity $\boldsymbol{\lambda}$ is constant and because we use the upwind numerical flux, the linear system is triangular. It can thus be solved cell by cell, simply sweeping the mesh in the direction of the velocity vector.

More precisely, expression (8.11) shows that the solution f^n depends only on the values of f^n in cell L and in the upwind cells. For a given velocity v we can build a dependency graph. The vertices of the graph are associated to cells and the edges to the cells interfaces or boundaries. We consider two fictitious additional vertices: the "upwind" vertex and the "downwind" vertex. The dependency graph for a simple unstructured mesh and a given constant velocity is represented in Figure 8.3. The construction can be generalized to any unstructured mesh with flat faces. This flatness condition ensures that the kinetic velocity crosses the faces in only one direction. This ensures that the graph does not contain loops. For more details, we refer to [7].

For solving one transport equation for a given constant velocity $\boldsymbol{\lambda}$, the algorithm is then the following:

(i) First we perform a topological ordering of the dependency graph.

(ii) First time-step: Assembly, LU decomposition and storage of the local cell matrices.

These computations can also be redone during each time-step for saving memory (but it is more CPU demanding).

(iii) For each cell (in topological order):

(a) Compute volume terms.

(b) Compute upwind fluxes.

(c) Solve the local linear system.

(d) Extract the results to the downwind cells.

For more details on our implementation of the implicit Discontinuous Galerkin method, we refer to [7].

3.3 Parallelization

The whole transport solver is parallelized. We adopt different strategies for the parallelism in the poloidal directions and for the toroidal direction.

Poloidal parallelism: task graph.

Because of the dependency graph we cannot perform all the computations in parallel. For instance, for the mesh of Figure 8.3. it is necessary to compute the solution in Cell 9 first. But then Cells 8 and 10 can be computed independently in parallel, *etc.*

The parallelization is done by a task graph approach. We have tested several implementations. For more details on the implementation, we refer for instance to [7], where the algorithm is parallelized with the StarPU runtime, or to [18] where we use a specialized DAG (Direct Acyclic Graph) clustering algorithm. The DAG algorithm relies on an OpenMP implementation.

Toroidal parallelism: message passing

For the toroidal direction, we associate to each poloidal plane one MPI process. Thus, the shift simply consists in an MPI send/receive operation with the neighbor poloidal planes. This approach imposes a constraint on the time step. An alternative would be to replace the shift by a semi-Lagrangian solver. This would imply exchanges with more neighbors and thus more MPI communications.

4 Kinetic solver

In this section, we present the whole numerical scheme to solve the kinetic model (8.6) and we validate it on two and three-dimensional test cases.

4.1 Splitting

The kinetic model (8.6) is solved using a splitting method. Starting from f_k^n the kinetic fields at time step $t_n = n\Delta t$, we make the following two steps to update them:

(i) We first solve the n_v free transport equations

$$\partial_t f_k + \boldsymbol{\lambda}_k \cdot \boldsymbol{\nabla}_{\boldsymbol{r}} f_k = 0,$$

using the transport solvers introduced in Section 3. We denote by f_k^* the value of the fields after the free transport step (8.9).

(ii) then the relaxation to the equilibrium distribution, corresponding to the right-hand side of (8.6), is solved using an over-relaxation process as used in [50, 22, 28]. For obtaining the new value of the field at time step t_{n+1}, the kinetic fields f_k^* are recombined according to

$$f_k^{n+1} = \omega f_k^{*,eq} + (1-\omega) f_k^*. \tag{8.12}$$

where $\omega > 0$ is the numerical relaxation parameter. This scheme can be formally derived using a θ-scheme for the relaxation part of Equation (8.6) and this leads to the following relation: $\omega = \Delta t/(\tau + \theta \Delta t)$. This relation is actually not taken into account: the actual numerical scheme considers a given ω, independent from the time step and with no reference to the parameter τ. We refer to [24] for more details.

For $\omega = 1$, we obtained the classical transport-projection scheme. The over-relaxation formula (8.12) enables us to obtain second order accuracy in time when the relaxation parameter $\omega = 2$. When the relaxation parameter satisfies $1 \leqslant \omega < 2$, then the scheme is only first-order accurate and the time integration introduces a numerical damping. This damping can sometimes be used for numerical stabilization purpose.

4.2 CFL condition

As we are using an implicit transport solver in the poloidal plane, we want to emphasize that the time step is not constrained by the poloidal discretization or the poloidal speed λ_p. The only constraint comes from the choice of the shift transport solver in the toroidal direction which imposes the relation: $\lambda_t \Delta t = \Delta\varphi$. This implies a CFL bound for the toroidal dynamics:

$$n_{\mathrm{CFL}}^{\mathrm{tor}} = \frac{\lambda_t \Delta t}{\Delta \varphi} = 1. \tag{8.13}$$

However, the full CFL number is defined by

$$n_{\mathrm{CFL}}^{\mathrm{full}} = \frac{u_{\max} \Delta t}{\min(\delta_p, \Delta\varphi)} \tag{8.14}$$

where $\delta_p > 0$ denotes the minimal distance between two interpolation points in the poloidal plane and $u_{\max}$ the maximal advection speed over the simulation. Unlike explicit methods, this CFL number is not constrained by our scheme and as a consequence, δ_p can be set without stability considerations, i.e. independently of the possible advection speed in the toroidal direction.

4.3 Boundary conditions

In the φ direction, the boundary conditions are naturally periodic. At the poloidal planes boundaries, we consider Dirichlet boundary conditions. More precisely, in the DG scheme

(8.10) if R corresponds to a boundary, then the unknown value of f_R is given by

$$f_R = f^{eq}(\rho, \boldsymbol{u}),$$

where ρ and $\boldsymbol{u}$ are imposed boundary data and f^{eq} is given by (8.7).

4.4 Validation in two-dimensional geometry

First we validate the kinetic solver described above in a single poloidal plane. We consider the two-dimensional rotation of a Gaussian pulse. The pulse is given by

$$g(r, z, \varphi, t) = \exp(-30\left((r' - 1)^2 + z'^2 + \varphi'^2\right)),$$

with

$$\begin{aligned} r' &= \cos(\beta t) r + \sin(\beta t) z, \\ z' &= \cos(\beta t) z - \sin(\beta t) r, \\ \varphi' &= 0, \end{aligned}$$

and we take $\beta = 1/4$. This function satisfies the advection equation

$$\partial_t g + \boldsymbol{u} \cdot \boldsymbol{\nabla} g = 0,$$

with

$$\boldsymbol{u} = \beta \begin{pmatrix} -z \\ r \\ 0 \end{pmatrix},$$

or equivalently the conservative transport equation:

$$\partial_t g + \boldsymbol{\nabla} \cdot (g\boldsymbol{u}) = 0,$$

since the velocity field is divergence free: $\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0$. We solve this equation in the disk

$$\Omega = \left\{(r, z, \varphi), \quad (r - r_t)^2 + z^2 < 4, \quad \varphi = 0\right\}.$$

We numerically compute the above solution with two different meshes with refinement levels of 5 and 10 (see Figure 8.5) and with $n_t = 500$ and $n_t = 1000$ time steps. The time step is given by $\Delta t = 2\pi/n_t$. We take $\lambda_p = 1$, use polynomials of degree 2 in the DG method and $\omega = 2$. The numerical solution is plotted at time $t_{\max} = 2\pi$ on Figure 8.4. We observe that the Gaussian shape is well preserved by the LBM scheme. We also check the order of convergence in the L^2 norm. The measured numerical order is here 2.405, which is consistent with the theoretical order of convergence.

The time scheme is only second order accurate and is thus the limiting factor for the convergence. Anyway we also check the spatial convergence of the scheme for higher order DG approximation. For this, we use a sufficiently small time-step in such a way that the error due to the time approximation is negligible. We perform simulations with two meshes with two different refinements of 10 and 20 (see Figure 8.5). We then obtain the numerical orders of Table 8.1, which is still consistent with the theoretical order of convergence.

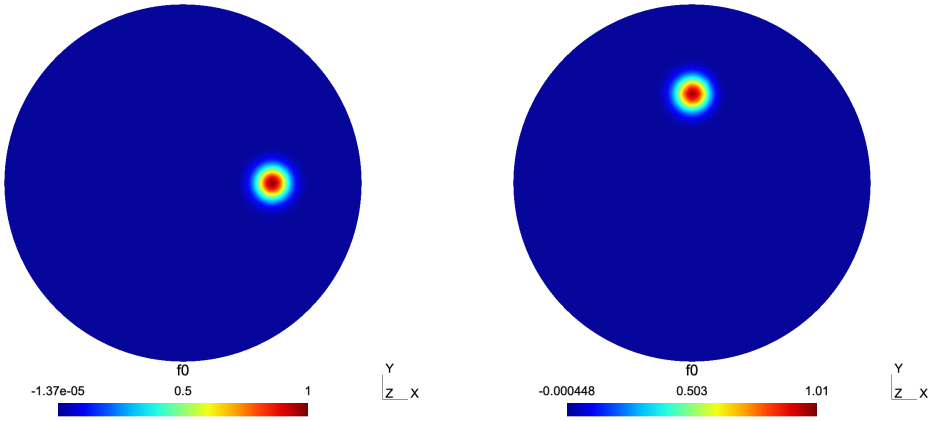

Figure 8.4: (Validation in two dimensions) Numerical solution at times $t = 0$ (left) and $t = 2\pi$ (right) after the Gaussian pulse has done a quarter turn. Parameters of the kinetic solver: $\Delta t = 2\pi/n_t$ with $n_t = 1000$, $\omega = 2$, $\lambda_p = 1$ and DG of order 2.

p	num. order
1	0.997
2	2.68
3	3.772

Table 8.1: (Validation in two dimensions) Numerical order of the DG scheme for several values of the polynomial order p. In this test, the time step is chosen in such a way that the time error is negligible. Parameters of the kinetic solver: $\lambda_p = 1$, $\omega = 2$.

4.5 Validation in a 3D periodic cylinder

Now we also activate transport in the third direction. We consider a three-dimensional helical shift of a Gaussian pulse. The pulse is given by

$$g(r, z, \varphi, t) = \exp(-30((r' - 1)^2 + z'^2 + \varphi'^2)),$$

with

$$\begin{aligned} r' &= \cos(2\pi\gamma t)r + \sin(2\pi\gamma t)z, \\ z' &= \cos(2\pi\gamma t)z - \sin(2\pi\gamma t)r, \\ \varphi' &= \varphi + \beta t. \end{aligned}$$

We can also consider a periodic function in the z direction

$$g(r, z, \varphi, t) = \exp(-30((r' - 1)^2 + z'^2)) \sin(\pi\varphi').$$

and we take $\gamma = 0.04$, $\beta = 0.25$. This function satisfies the transport equation

$$\partial_t g + \boldsymbol{\nabla_r} \cdot (g\boldsymbol{u}) = 0,$$

with the divergence free velocity field

$$\boldsymbol{u} = \begin{pmatrix} -2\pi\gamma z \\ 2\pi\gamma r \\ -\beta \end{pmatrix}.$$

The computational domain is the cylinder

$$\Omega = \left\{(r, z, \varphi), \quad (r - r_t)^2 + z^2 < 4, \quad -1 = \varphi_{\min} < \varphi < 1 = \varphi_{\max}\right\}.$$

We consider a maximal time $t_{\max} = 1$. In this way, the Gaussian pulse will move a little bit, without touching the boundaries in the φ direction. As stated above, parallelism is managed by OpenMP and MPI. OpenMP is used for optimizing the transport solver in the poloidal planes. MPI communications are used for the parallelism in the toroidal direction. We recall that parallelism in the φ direction is directly linked to the number of poloidal planes, as one MPI process is associated to each plane. If, for instance, we choose $n_p = 64$ toroidal planes then the space step in the φ direction equals

$$\Delta\varphi = \frac{\varphi_{\max} - \varphi_{\min}}{n_p},$$

and there are n_p MPI processes. Moreover, with a kinetic speed $\lambda_t = 1$ and as we are considering LBM in the φ direction, the time step is given by

$$\Delta t = \frac{\Delta\varphi}{\lambda_t} = 0.03125.$$

We thus have to perform n_t time iterations

$$n_t = \frac{t_{\max}}{\Delta t} = 32.$$

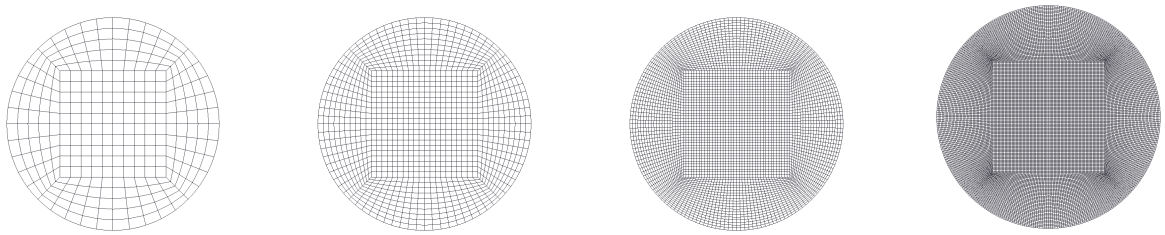

Figure 8.5: Poloidal meshes with refinements $n_{\text{raf}} = 5$, 10, 20 and 40. The disk is decomposed into 12 patches and each patch is meshed with n_{raf}^2 quadrilaterals, where n_{raf} is the refinement number.

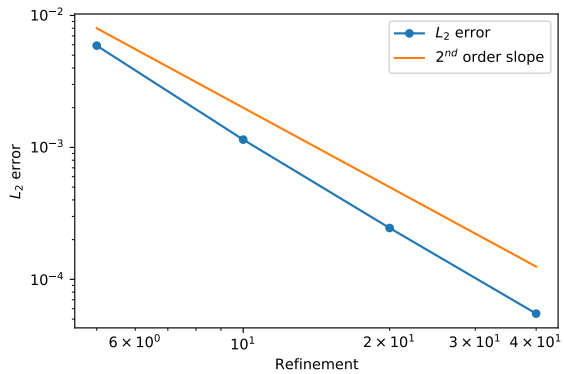

Figure 8.6: (Validation in three dimensions) L^2 error as a function of the refinement level. Parameters: $\lambda_t = \lambda_p = 1$, DG order $p = 2$.

In order to check the convergence order, we have considered three different meshes of the poloidal plane with a refinement of 5, 10, 20 and 40, as represented in Figure 8.5. While refining in the poloidal direction, we also refine in the toroidal direction by taking 32, 64, 128, 256 poloidal planes or equivalently MPI processes. We consider the second order DG scheme and we take $\lambda_p = \lambda_t = 1$. We compute the error in the L^2 norm and obtain the error curve of Figure 8.6. The order of convergence based on the 64 and 128 refinement levels is 2.226 (2.155 for 128-256 refinements). In this way, we have numerically validated the expected accuracy of the full kinetic scheme.

5 Applications to plasma dynamics

In this section, we apply the transport solver to more complex test cases from plasma physics. We consider the transport of charged particles in a tokamak along the magnetic field lines. These charged particles create an electric field and the drift theory then implies that, in the mean, their motion is perturbed by this electric field.

For validation purpose, we neglect curvature effect and thus consider transport models in **straight toroidal domains**. Extension to curved domains will be studied in a future work.

Moreover, next simulations consider **annular poloidal planes**, but there is no restriction on the geometry of the poloidal planes. This annular geometry is chosen in order to validate the model on classical test cases.

5.1 Drift-kinetic like model

Here we consider the following simplified model. The charge density $\rho(\boldsymbol{r}, t)$ is transported according to equation (8.2) at the drift velocity

$$\boldsymbol{u} = \boldsymbol{E} \times \boldsymbol{e}_\varphi + \boldsymbol{B}, \tag{8.15}$$

where the electric field $\boldsymbol{E}$ is the gradient of the electric potential V,

$$\boldsymbol{E} = -\nabla_{\boldsymbol{r}} V. \tag{8.16}$$

and $\boldsymbol{B}$ denotes a magnetic field that is externally imposed, $\boldsymbol{B} = (b_r, b_z, b_\varphi)^T$. In other words, the drift velocity writes:

$$\boldsymbol{u} = (-\frac{\partial V}{\partial z} + b_r, \frac{\partial V}{\partial r} + b_z, b_\varphi)^T.$$

The electric potential, whose derivative in the φ direction is neglected, is a solution to a Poisson equation

$$-\Delta V = \rho, \tag{8.17}$$

where the Laplacian operator acts only in the poloidal plane:

$$\Delta = \frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial z^2}.$$

We assume homogeneous Dirichlet boundary conditions for the potential. This implies that $\boldsymbol{E} \times \boldsymbol{e}_\varphi$ is tangent at the boundary. If $\boldsymbol{B}$ is also tangent to the boundary, then no mass escapes the domain. This model, made of the transport equation along $\boldsymbol{B}$ and the poloidal drift is a very simplified drift-kinetic model with a unique parallel velocity [51]. Thus we focus on the resolution of the transport dynamics in the spatial domain and do not consider physical kinetic effect.

From (8.15), (8.16) and (8.17) we see that the drift velocity is obtained from a linear operator applied to the density

$$\boldsymbol{u} = \boldsymbol{u}(\rho).$$

The drift-kinetic model can also be written

$$\partial_t \rho + \nabla_{\boldsymbol{r}} \cdot (\rho \boldsymbol{u}(\rho)) = 0, \tag{8.18}$$

and therefore is a non-linear model.

In addition to the transport solver described above, we thus have to implement a poloidal Poisson solver, which will be applied in each poloidal plane independently. We will test the accuracy of the coupling between the transport and the Poisson solver. In practice, we use a standard conforming finite element solver for the Poisson equation. The solver uses the same mesh and the same order of basis functions as the transport solver. Once the Poisson

equation is solved, the electric potential is computed according to (8.16). The numerical electric field is thus generally time and space dependent and discontinuous at the interfaces between the DG cells.

Given a stationary solution to the drift-kinetic model, a slight perturbation is applied to this particular solution. In certain configurations one observes instabilities. The objective of the following two test cases is to compute numerically the instability rate. In simple geometries the instability rate can be evaluated by analytical or semi-analytical methods. It is then possible to test the accuracy of the Lattice-Boltzmann approach. We first consider a two-dimensional diocotron instability and then a fully three-dimensional instability.

5.2 Two-dimensional diocotron test-case

We suppose that $\boldsymbol{B} = (0,0,0)^T$. Then the solution does not depend on φ and we recover the so-called guiding-center model. The problem is thus indeed two-dimensional. We note $\alpha \in \mathbb{R}^+$ and $\theta \in [0, 2\pi)$, the polar coordinates in the poloidal plane:

$$\begin{aligned} r &= \alpha \cos\theta + r_t, \\ z &= \alpha \sin\theta, \end{aligned}$$

with $r_t > 0$ the radius of the torus. Here the computational domain is a circular annulus defined by

$$\Omega = \left\{(r, z, \varphi), \varphi = 0 \text{ and } \alpha_{\min} \leqslant \alpha \leqslant \alpha_{\max}\right\}.$$

We will test our solver on the Diocotron test case described in [26, 67]. The density is initialized with a continuous function

$$\rho(r, z, \varphi, t) = \tilde{\rho}(\alpha, \theta, t) = (1 + \varepsilon \cos(k\theta))\, e^{-\frac{(\alpha-\alpha_0)^2}{2\sigma^2}}, \tag{8.19}$$

with $\sigma > 0$ and $\alpha_0 \in (\alpha_{\min}, \alpha_{\max})$ and where $\varepsilon > 0$ and $k \in \mathbb{N}^*$ parametrize the perturbation. The electric potential V satisfies equation (8.17) with homogeneous Dirichlet boundary conditions at $\alpha = \alpha_{\min}$ and $\alpha = \alpha_{\max}$.

For $\varepsilon = 0$ this density is a solution to the drift kinetic model (8.18): indeed the density, and thus also the potential, has a cylindrical symmetry. Then, because of (8.15), the velocity $\boldsymbol{u}$ is parallel to the vector $(-z, r)$ and the transport equation leaves the initial density invariant.

When $\varepsilon > 0$, an instability may appear to depend on the parameters $\alpha_{\min}$, $\alpha_{\max}$ and α_0. In this special case, where the initial condition has cylindrical symmetry, we can estimate the instability growth rate with a simpler one-dimensional numerical method in order to compare it with the results of our solver. We explain now this simpler method.

Instability rate

In polar coordinates in the poloidal plane, the guiding-center model writes

$$\begin{aligned} \partial_t \tilde{\rho} + \frac{1}{\alpha}\left(\partial_\alpha \tilde{V} \partial_\theta \tilde{\rho} - \partial_\theta \tilde{V} \partial_\alpha \tilde{\rho}\right) &= 0, \\ -\frac{1}{\alpha}\partial_\alpha\left(\alpha\, \partial_\alpha \tilde{V}\right) - \frac{1}{\alpha^2}\partial_{\theta^2}\tilde{V} &= \tilde{\rho}. \end{aligned} \tag{8.20}$$

Let $(\tilde{\rho}_0(\alpha), \tilde{V}_0(\alpha))$ be a stationary radial solution to this system. We look for a solution which writes as the sum of the stationary solution and a small perturbation

$$\tilde{\rho}(\alpha,\theta,t) = \tilde{\rho}_0(\alpha) + \varepsilon\, \tilde{\rho}_k(\alpha)\, e^{ik\theta}\, e^{-i\eta t},$$
$$\tilde{V}(\alpha,\theta,t) = \tilde{V}_0(\alpha) + \varepsilon\, \tilde{V}_k(\alpha)\, e^{ik\theta}\, e^{-i\eta t}.$$

Inserting this ansatz into system (8.20), we obtain

$$-\eta\tilde{\rho}_k(\alpha) + \frac{k}{\alpha}\left(\tilde{V}_0'(\alpha)\tilde{\rho}_k(\alpha) - \tilde{V}_k(\alpha)\tilde{\rho}_0'(\alpha)\right) = 0, \tag{8.21}$$

$$-\tilde{V}_k''(\alpha) - \frac{1}{\alpha}\tilde{V}_k'(\alpha) + \frac{k^2}{\alpha^2}\tilde{V}_k(\alpha) = \tilde{\rho}_k(\alpha). \tag{8.22}$$

Plugging (8.22) into (8.21), this system can be rewritten as the following equation

$$\left(\frac{1}{\alpha}\tilde{V}_0'(\alpha) - c\right)\left(-\tilde{V}_k''(\alpha) - \frac{1}{\alpha}\tilde{V}_k'(\alpha) + \frac{k^2}{\alpha^2}\tilde{V}_k(\alpha)\right) = \frac{1}{\alpha}\tilde{V}_k(\alpha)\tilde{\rho}_0'(\alpha), \tag{8.23}$$

which is a generalized eigenvalue problem with eigenvalue η/k. We keep the eigenvalue with the largest imaginary part (if there exists) and the instability rate is then given by $\Im(\eta)$.

In the case (8.19), the radial solution is

$$\tilde{\rho}_0(\alpha) = e^{-\frac{(\alpha-\alpha 0)^2}{2\sigma^2}}.$$

Moreover, since $\tilde{V}_0'$ is solution to the following elliptic equation

$$-\tilde{V}_0''(\alpha) - \frac{1}{\alpha}\tilde{V}_0'(\alpha) = \tilde{\rho}_0(\alpha),$$

we obtain the following expression

$$\tilde{V}_0'(\alpha) = \frac{1}{\alpha}\left(c_1 - \sqrt{\frac{\pi}{2}}\alpha_0\sigma\,\mathrm{erf}\left(\frac{\alpha-\alpha_0}{\sqrt{2}\sigma}\right) + \sigma^2 e^{-\frac{(\alpha-\alpha 0)^2}{2\sigma^2}}\right),$$

and then

$$\tilde{V}_0(\alpha) = \int_{\alpha_{\min}}^{\alpha} \tilde{V}_0'(\alpha)d\alpha + c_2.$$

The Dirichlet boundary condition $\tilde{V}_0(\alpha_{\min}) = 0$ implies that $c_2 = 0$, and $\tilde{V}_0(\alpha_{\max}) = 0$ implies that $\int_{\alpha_{\min}}^{\alpha_{\max}} \tilde{V}_0'(\alpha)d\alpha = 0$. Therefore, we can approximate c_1 with a numerical integration.

For finding out the unstable mode, we discretize the one-dimensional equation (8.23) using a finite difference method over the interval $[\alpha_{\min}, \alpha_{\max}]$. We approximate the first and second derivatives of $\tilde{V}_k(\alpha)$ by standard central differences. Finally, the solution of the equation (8.23) is numerically solved as a generalized eigenvalue problem. Of course, the number of discretization points is taken large enough, in order to achieve high accuracy.

Validation

We choose $\alpha_{\min} = 1$, $\alpha_{\max} = 10$, and the following parameters for the initial condition (8.19): $\alpha_0 = 4.5$, $\sigma = 0.5$, and $k = 2$. Using the method described in the above section, we obtain that this initial condition (8.19) leads to an instability of rate 0.15215.

To validate our code, we have done a dynamic numerical simulation with the full kinetic solver using the same parameters. We consider $\lambda_p = 7$, use polynomials of degree 2 in the DG method, a time step $\Delta t = 0.0125$ and a relaxation parameter $\omega = 1.999$. We consider a mesh of size $N_\alpha \times N_\theta = 100 \times 60$. Density plots are given in Figure 8.7. The execution takes approximately 1.5 hours. We define the poloidal CFL number by

$$n_{\mathrm{CFL}}^{\mathrm{pol}} = \frac{v_{\max}\Delta t}{\delta_p},$$

with v_{max} the maximal speed and δ_p the minimal distance between two interpolation points. In this case, the CFL number equals 2.35. With such a CFL number, the calculation would be unstable with a DG explicit method. In Figure 8.8, we plot the k-th Fourier mode in θ of the potential whose formula is given by:

$$h(t) = \frac{1}{(\alpha_{\max} - \alpha_{\min})} \int_{\alpha_{\min}}^{\alpha_{\max}} \int_0^{2\pi} e^{-ik\theta} \tilde{V}(\alpha, \theta, t) d\alpha d\theta \tag{8.24}$$

and estimate the slope in the linear growth regime. We observe an instability rate of 0.15123. This is in accordance with the theoretical instability rate.

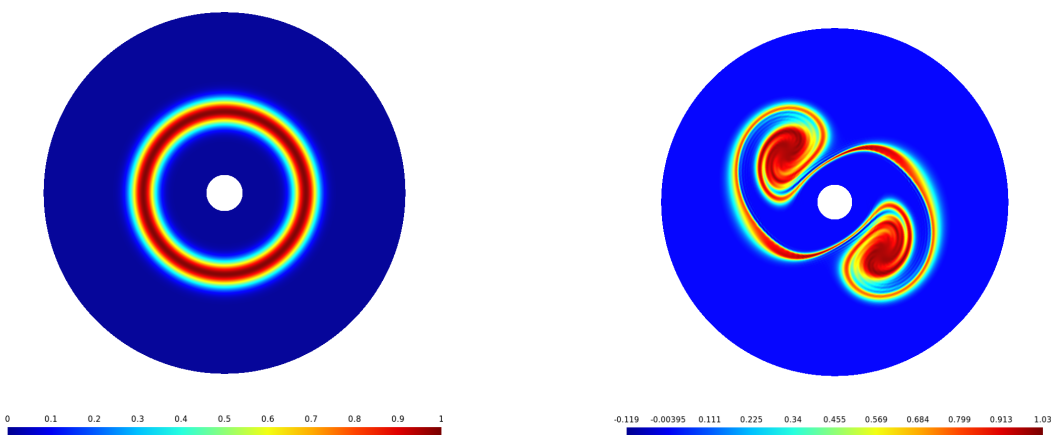

Figure 8.7: (Two-dimensional diocotron test case) Density at $t = 0$ and $t = 100$. Parameters: $\lambda_p = 7$, DG order $p = 2$, $\Delta t = 0.0125$, $\omega = 1.999$.

5.3 Diocotron test-case in a periodic cylinder

We consider now the three dimensional drift kinetic model (8.2)-(8.15)-(8.16)-(8.17) We choose a magnetic field $\boldsymbol{B}$ oriented in the θ and φ directions

$$\boldsymbol{B} = (-b_\theta \sin(\theta), b_\theta \cos(\theta), b_\varphi)^T,$$

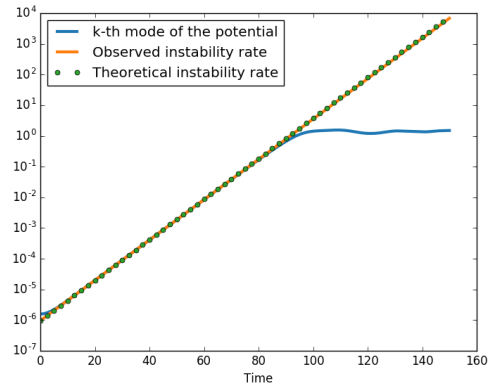

Figure 8.8: (Two-dimensional diocotron test case) k-th Fourier mode of the potential over time and the instability rate given by the slope of the Fourier mode between $t = 20$ and $t = 60$, compared to the theoretical instability rate. Parameters: $\lambda_p = 7$, DG order $p = 2$, $\Delta t = 0.0125$, $\omega = 1.999$.

with $b_\theta > 0$ and $b_\varphi > 0$. The computational domain is the cylinder

$$\Omega = \left\{ (r, z, \varphi) \;\middle|\; \alpha_{\min} \leqslant \sqrt{(r - r_t)^2 + z^2} \leqslant \alpha_{\max},\ 0 \leqslant \varphi \leqslant L \right\}.$$

We consider periodic boundary conditions in φ and homogeneous Dirichlet boundary conditions for the potential $\tilde{V}$. We consider the following initial density

$$\rho(r, z, \varphi, 0) = \tilde{\rho}(\alpha, \theta, \varphi, 0) = \left(1 + \varepsilon \cos\left(k\theta + \ell\varphi \frac{2\pi}{L}\right)\right) e^{-\frac{(\alpha - \alpha_0)^2}{2\sigma^2}}.$$

with $\sigma > 0$ and $\alpha_0 \in (\alpha_{\min}, \alpha_{\max})$ and where $\varepsilon > 0$ and $k, \ell \in \mathbb{N}^*$ parametrize the perturbation. Like in the two-dimensional case, we first derive the eigenvalue problem for calculating the instability rate. We show that this rate does not depend on the perturbation along φ.

Instability rate

With cylindrical coordinates, the system (8.1)-(8.16)-(8.17) becomes

$$\partial_t \tilde{\rho} - \frac{1}{\alpha} \partial_\theta \tilde{V}\ \partial_\alpha \tilde{\rho} + \left(\frac{1}{\alpha} \partial_\alpha \tilde{V} + b_\theta\right) \partial_\theta \tilde{\rho} + b_\varphi \partial_\varphi \tilde{\rho} = 0, \tag{8.25}$$

$$-\frac{1}{\alpha} \partial_\alpha (\alpha \partial_\alpha \tilde{V}) - \frac{1}{\alpha^2} \partial_{\theta^2} \tilde{V} = \tilde{\rho}. \tag{8.26}$$

Let $(\tilde{\rho}_0(\alpha), \tilde{V}_0(\alpha))$ be a stationary radial solution of this system, which only depends on α and is constant along the θ and z directions. We look for the equation satisfied by a small perturbation around this stationary solution

$$\tilde{\rho}(\alpha, \theta, \varphi, t) = \tilde{\rho}_0(\alpha) + \varepsilon\ \tilde{\rho}_{k,l}(\alpha)\ e^{ik\theta} e^{i\ell\varphi} e^{-i\eta t},$$
$$\tilde{V}(\alpha, \theta, \varphi, t) = \tilde{V}_0(\alpha) + \varepsilon\ \tilde{V}_{k,l}(\alpha)\ e^{ik\theta} e^{i\ell\varphi} e^{-i\eta t},$$

with $\varepsilon > 0$. We plug this expression into Equations (8.25)-(8.26) and keep only the $O(\varepsilon)$ terms. We obtain

$$-\eta\tilde{\rho}_{k,\ell}(\alpha) - \frac{k}{\alpha}\tilde{\rho}_0'(\alpha)\tilde{V}_{k,\ell}(\alpha) + k\tilde{\rho}_{k,\ell}(\alpha)\left(\frac{1}{\alpha}\tilde{V}_0'(\alpha) + b_\theta\right) + \ell\tilde{\rho}_{k,\ell}(\alpha)b_\varphi = 0, \tag{8.27}$$

$$-\frac{1}{\alpha}\partial_\alpha(\alpha\tilde{V}_{k,\ell}'(\alpha)) + \frac{k^2}{\alpha^2}\tilde{V}_{k,\ell}(\alpha) = \tilde{\rho}_{k,\ell}(\alpha). \tag{8.28}$$

Plugging (8.28) into (8.27), we finally get

$$\left(\frac{1}{\alpha}\tilde{V}_0'(\alpha) + b_\theta + \frac{\ell}{k}b_\varphi - \frac{\eta}{k}\right)\left(-\tilde{V}_{k,\ell}''(\alpha) - \frac{1}{r}\tilde{V}_{k,\ell}'(\alpha) + \frac{k^2}{\alpha^2}\tilde{V}_{k,\ell}(\alpha)\right) = \frac{1}{\alpha}\tilde{\rho}_0'(\alpha)\tilde{V}_{k,\ell}(\alpha). \tag{8.29}$$

which is the same equation obtained in (8.23) for the two-dimensional case, except for the presence of the term $b_\theta + \frac{\ell}{k}b_\varphi$. Consequently, compared to the 2D case, the eigenvalues are only shifted by this real value and the instability rate does not change as it is equal to the largest imaginary part of the eigenvalues.

Validation

The poloidal planes have the same dimension as in the two-dimensional test case ($\alpha_{\min} = 1$, $\alpha_{\max} = 10$) while the cylinder length is taken equal to $L = 200$. The magnetic field is supposed to have a large toroidal component compared with its poloidal ones: $b_\theta = 0.1$ and $b_\varphi = 200$. The distribution is initially concentrated around $\alpha_0 = 4.5$ with a Gaussian distribution of standard deviation $\sigma = 0.5$. Finally, the perturbation is taken of size $\varepsilon = 10^{-6}$ and involves modes $k = 2$ in the θ direction and $\ell = 1$ in the φ direction.

The numerical parameters of the simulations are as follows. Each poloidal mesh is composed of $4 \times N_\alpha \times N_\theta = 4 \times 80 \times 50$ elements. In the toroidal direction, we use $n_p = 128$ poloidal planes and so 128 MPI processes. Consistently with the involved advection velocities, we choose kinetic speeds of norm $\lambda_p = 7$ in the poloidal plane and $\lambda_t = 600$ in the toroidal direction. The time step is thus equal to $\Delta t = L/(n_p\lambda_t) = 0.002604$ and the relaxation parameter is chosen equal to $\omega = 1.99$.

Figure 8.9 shows the time evolution of the density and Figure 8.10 the density at final time $t_{\max} = 100$ for different poloidal planes. As expected, we can observe that the density in the poloidal planes are identical up to a rotation. As time goes on, two vortices develop and lead to fine structures in the poloidal planes as in the two-dimensional case. The computation takes 26 hours.

We would like to emphasize that the full CFL number, defined in (8.14), takes the value 33.17. This is due to the very fast speed in the toroidal direction which is not resolved by the poloidal plane mesh. Indeed, the minimal distance between interpolation points in the poloidal planes equals $\delta_p = 0.01570$ and the time step is not small enough to capture speed of order 200 for this fine resolution. An explicit scheme would require either decreasing the time step and thus increasing the toroidal discretization, or a coarser poloidal mesh at the cost of a loss of accuracy. Consequently, explicit schemes would not be able to cope with both the large velocity in the toroidal direction and a very fine grid in the poloidal plane.

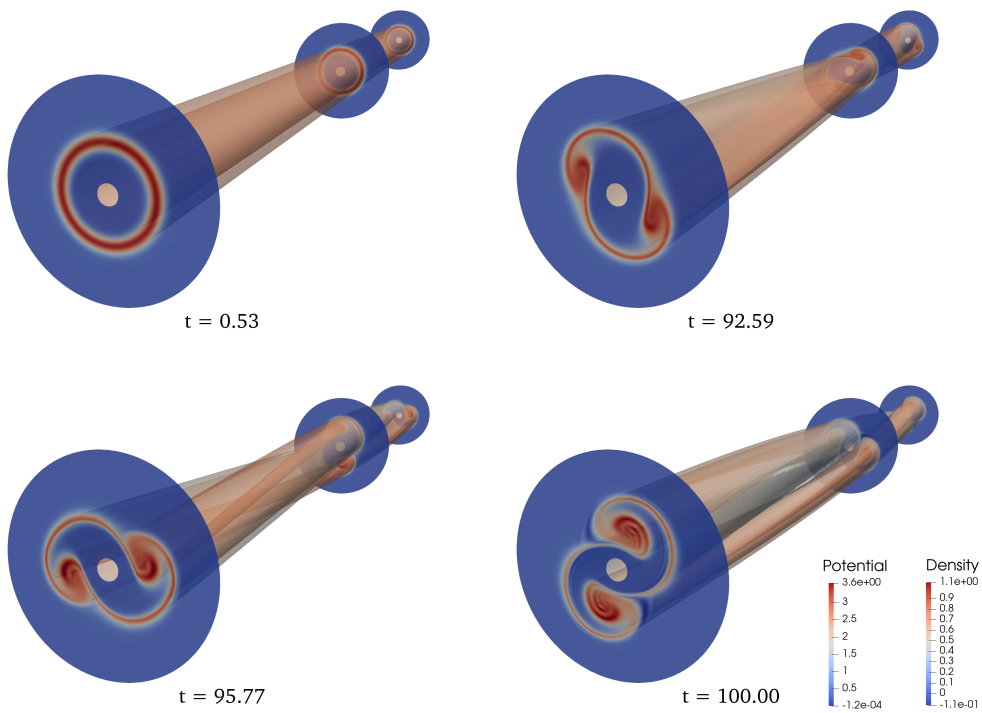

Figure 8.9: (Three-dimensional diocotron test case) Four successive instants of the 3D solution showing cross sections of density colormaps and density isosurface colored by the potential. Parameters: $\lambda_p = 7$, $\lambda_t = 600$, DG order $p = 2$, $\omega = 1.99$, $n_p = 65$.

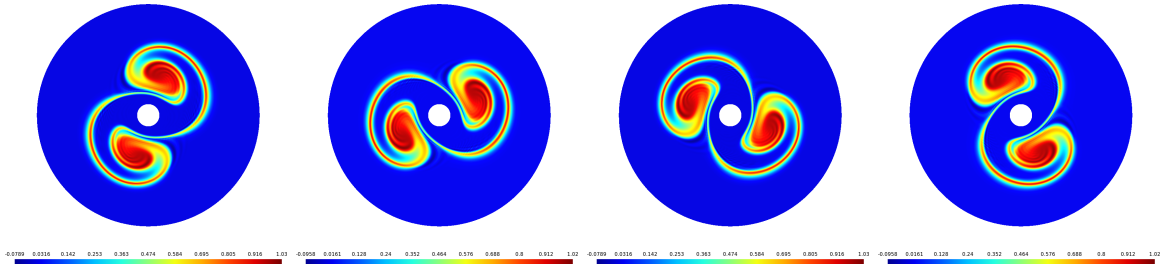

Figure 8.10: (Three-dimensional diocotron test case) Density at $t_{\max} = 100$ for the poloidal planes $\varphi = 0$, $\varphi = \frac{L}{4}$, $\varphi = \frac{L}{2}$ and $\varphi = \frac{3L}{4}$. Parameters: $\lambda_p = 7$, $\lambda_t = 600$, DG order $p = 2$, $\omega = 1.99$, $n_p = 128$.

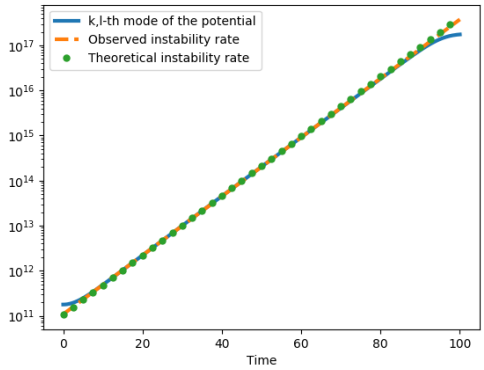

Figure 8.11: (Three-dimensional diocotron test case) Instability rate observed compared to the theoretical one. Parameters: $\lambda_p = 7$, $\lambda_t = 600$, DG order $p = 2$, $\omega = 1.99$, $n_p = 128$.

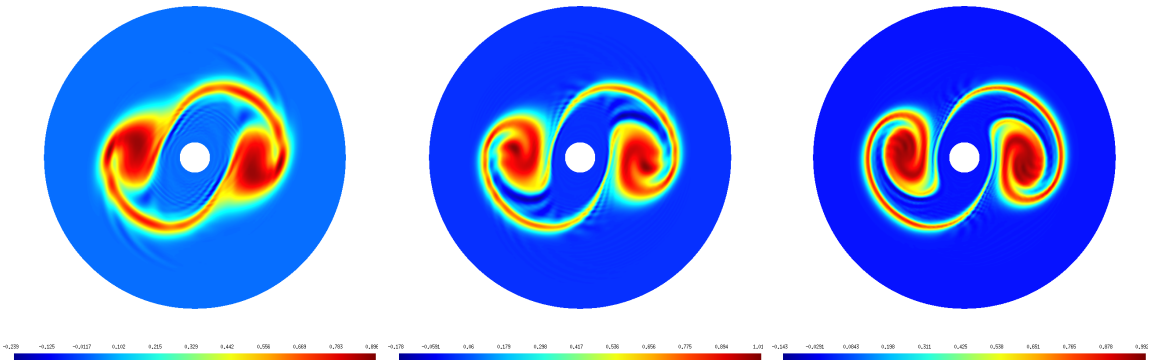

Figure 8.12: (Three-dimensional diocotron test case) Density at $t_{max} = 100$ in the poloidal plane $\varphi = 0$ obtained with $n_p = 32$, $n_p = 64$ and $n_p = 128$. Parameters: $L = 1$, $\lambda_p = 7$, $\lambda_t = 3$, DG order $p = 2$, $\omega = 1.99$.

In Figure 8.11, we plot the time evolution of quantity (8.24) integrated along φ. We compare the theoretical instability rate computed before (see Sec. (5.3)) to the one obtained by the three-dimensional kinetic solver. We expect an instability rate of 0.15215, and we obtain 0.15061, which again validates the kinetic solver.

When we decrease the number of MPI processes n_p, the number of planes in the poloidal direction decrease and the poloidal discretization step $\Delta\varphi$ is then smaller. Figure 8.12 shows the densities obtained at time $t_{max} = 100$ for the poloidal plane $\varphi = 0$ for $n_p = 32$, $n_p = 64$ and $n_p = 128$. As the number of MPI processes decreases, we observe that the solution is less and less precise and dispersion effect are more and more present. This is mainly due to the increase of the time step imposed by the Lattice-Boltzmann method. Other toroidal solver will be studied in future works.

5.4 Parallel efficiency: weak scaling

Now, we want to compare the time of execution for different numbers of MPI processes n_p. We keep the parameters used in the section 5.3, we only reduce the final time to $t_{max} = 1$, in

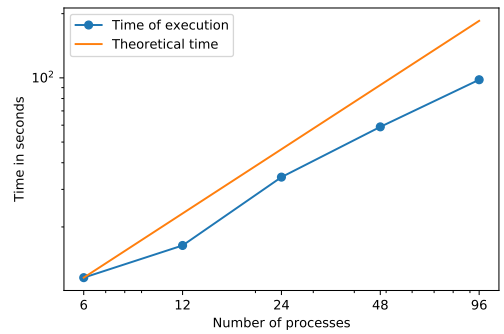

Figure 8.13: (Parallel weak scaling) Time of execution according to the number of MPI processes.

order to get shorter computations. As we have $\Delta t = \frac{L}{n_p \lambda_t}$, if the MPI communication time is neglected, we expect the time to double when the number of MPI processes is multiplied by 2. We perform numerical computations with $n_p = 6$, 12, 24, 48 and 96 MPI processes. We obtain the CPU time evolution of Figure 8.13. We observe a correct scaling of the parallel method when the number of MPI processes increases. We emphasize that for this test we have deactivated the OpenMP acceleration on each MPI node. For harnessing the OpenMP acceleration, we should run the code on a larger parallel computer.

6 Conclusion

In this work, we have proposed a new optimized numerical method for solving non-homogeneous conservative transport equations in toroidal geometries. The method is conservative, high-order in space and time and has the complexity of a time-explicit scheme. It is also able to handle unstructured meshes of the poloidal plane, which is very useful for numerical simulations in tokamaks, like ITER. Finally, it presents many features that allow an efficient parallelization.

The method has been first validated on academic transport test cases. The method has then been applied to more physical configurations. The transport has been coupled with a toroidal Poisson solver for computing the electric potential generated by the charge density motion. In this framework, we have been able to validate the method on the estimation of diocotron instability rates.

We are currently working on the extension of the method to other physical models: with more realistic toroidal geometries, more complex transport models, several populations of particles, and richer gyrokinetic models. We also plan to run the solver on larger supercomputers.

Appendix A. Subcharacteristic stability condition

We consider the kinetic model (8.6). The sub-characteristic stability condition is a stability condition associated to the equivalent equation of (8.6) in the limit $\tau \to 0$. Indeed, in that

asymptotic, the equivalent equation writes:

$$\partial_t \rho + \nabla_{\boldsymbol{r}} \cdot (\rho \boldsymbol{u}) = \tau \, \nabla_{\boldsymbol{r}} \cdot (\mathcal{D} \nabla_{\boldsymbol{r}} \rho) + O(\tau^2),$$

where the $O(\tau)$ term involves a diffusion matrix given by:

$$\mathcal{D} = \begin{pmatrix} \frac{\lambda_p^2}{3} - u_r^2 & -u_r u_z & -u_r u_\varphi \\ -u_r u_z & \frac{\lambda_p^2}{3} - u_z^2 & -u_z u_\varphi \\ -u_r u_\varphi & -u_z u_\varphi & \frac{\lambda_t^2}{3} - u_\varphi^2 \end{pmatrix}.$$

We refer to [13] for the derivation of this equivalent equation. For this equation to be L^2 stable, this diffusion matrix has to be positive definite: this is the sub-characteristic condition. Here, the eigenvalues of $\mathcal{D}$ are

$$d_1 = \frac{\lambda_p^2}{3}, \quad d_2 = \frac{1}{2}\left(-\alpha + \frac{\lambda_p^2 + \lambda_t^2}{3} - \|\boldsymbol{u}\|^2\right), \quad d_3 = \frac{1}{2}\left(\alpha + \frac{\lambda_p^2 + \lambda_t^2}{3} - \|\boldsymbol{u}\|^2\right),$$

with

$$\alpha = \sqrt{\left(\frac{\lambda_p^2 + \lambda_t^2}{3} - \|\boldsymbol{u}\|^2\right)^2 + 4\frac{\lambda_p^2}{3}\left(u_\varphi^2 - \frac{\lambda_t^2}{3}\right) + 4\frac{\lambda_t^2}{3}\left(u_r^2 + u_z^2\right)}.$$

As d_1 is always positive and $d_2 \leqslant d_3$, the model is stable if and only if $d_2 \geqslant 0$ or equivalently

$$\frac{\lambda_p^2 + \lambda_t^2}{3} - \|\boldsymbol{u}\|^2 \geqslant \alpha$$

After some easy computations, we find that this is also equivalent

$$\frac{u_r^2}{\lambda_p^2} + \frac{u_z^2}{\lambda_p^2} + \frac{u_\varphi^2}{\lambda_t^2} \leqslant \frac{1}{3},$$

which states that all the velocity field $\boldsymbol{u}$ have to belongs to the ellipsoid with parameters $\left(\frac{\lambda_p}{\sqrt{3}}, \frac{\lambda_p}{\sqrt{3}}, \frac{\lambda_t}{\sqrt{3}}\right)$.

Chapter 9

Optimization of spatial control strategies for population replacement, application to *Wolbachia*

This chapter is totally independant of the rest of the thesis. It presents a work carried out in collaboration with Michel Duprez, Yannick Privat and Nicolas Vauchelet. It has been published in ESAIM: Control, Optimisation and Calculus of Variations, in an article entitled *Optimization of spatial control strategies for population replacement, application to Wolbachia*.

Abstract: In this article, we are interested in the analysis and simulation of solutions to an optimal control problem motivated by population dynamics issues. In order to control the spread of mosquito-borne arboviruses, the population replacement technique consists in releasing into the environment mosquitoes infected with the *Wolbachia* bacterium, which greatly reduces the transmission of the virus to the humans. Spatial releases are then sought in such a way that the infected mosquito population invades the uninfected mosquito population. Assuming very high mosquito fecundity rates, we first introduce an asymptotic model on the proportion of infected mosquitoes and then an optimal control problem to determine the best spatial strategy to achieve these releases. We then analyze this problem, including the optimality of natural candidates and carry out first numerical simulations in one dimension of space to illustrate the relevance of our approach.

1 Introduction and state of the art

Aedes mosquitoes are the main vector of the transmission to human of many diseases, such as dengue, zika, or chikungunya. Since there are still no vaccines against these diseases, the best way to fight against them is to act on the vector population. Several techniques have been proposed. Some approaches aim at reducing the size of the population of mosquitoes. The use of insecticides is one of them, but its environmental consequences are too important to be used for a long time and on a large scale. The sterile insect technique (SIT) or the incompatible insect technique (IIT) are very promising strategies, consisting in massive

releases of sterile or incompatible males, after mating with these males, the wild females will not produce viable eggs which should reduce the size of the populations (see e.g. [41] and references therein). This method has already been implemented successfully on the field (see [86, 98]). Other strategies are based on genetic manipulations like, for example, the release of insects carrying a dominant lethal (RIDL) [92, 55, 48].

However, the suppression of one population of insects might have consequences on the environment. Then, other approaches aim at replacing the wild population of mosquitoes by another population inoffensive to human. One strategy under investigation consists in using the bacteria *Wolbachia* taking advantage of phenomena called *cytoplasmic incompatibility* (CI) and *pathogen interference* (PI) [17, 84]. In key vector species such as *Aedes aegypti*, if a male mosquito infected with *Wolbachia* mates with a non-infected female, the embryos die early in development [96]. This is the so-called *cytoplasmic incompatibility* (CI). Moreover, it has been observed that *Aedes* mosquitoes infected with some *Wolbachia* strains are not able to transmit viruses like dengue, chikungunya and zika [94], this is the *pathogen interference* (PI). Then, one may release mosquitoes artificially infected by *Wolbachia* to mate with wild ones. Over time and if the releases are large and long enough, it can be expected that the majority of mosquitoes will carry *Wolbachia*, due to cytoplasmic incompatibility. As a result of PI, the mosquito population then has reduced vectorial competence.

In this paper, we focus on the *Wolbachia* strategy and investigate the question of optimizing the spatial distribution of the releases. Several mathematical models have been proposed for the *Wolbachia* technique, see e.g. [43, 44, 81, 60]. In these papers, the authors model the time dynamics of the mosquitoes population. Then, the question of optimizing the time of releases has been investigated e.g. in [19, 4, 12, 2]. However the spatial distribution of mosquitoes may have an impact on the success of the strategy. It is therefore relevant to add spatial dependence in mathematical models, which makes the study much more complicated.

In order to have a model simple enough to be tractable from a mathematical point of view, the authors in [8] introduce a model focusing only on the proportion of *Wolbachia*-infected mosquitoes, denoted p in the sequel :

$$p := \frac{n_{in}}{n_{in} + n_{un}}$$

where n_{in} is the density of *Wolbachia*-infected mosquitoes and n_{un} the density of uninfected mosquitoes. This quantity solves a scalar reaction-diffusion equation

$$\frac{\partial p}{\partial t} - D\Delta p = f(p),$$

where D is a diffusion coefficient and f is a bistable function[1]. For this model, the conditions to initiate the spatial spread are well-known [90]. It has been proved later in [89] that this model may be rigorously derived from a more general system governing the dynamics of *Wolbachia*-infected and *Wolbachia*-uninfected mosquitoes by performing a large fecundity asymptotics.

[1]The wording "**bistable function**" means that $f(0) = f(1) = 0$ and there exists $\theta \in (0,1)$ such that $f(x)(x-\theta) < 0$ on $(0,1) \setminus \{\theta\}$ (in particular, one has necessarily $f(\theta) = 0$ whenever f is continuous)

In this study, we are investigating the question of the best spatial strategy for mosquito release, i.e., giving a certain amount of mosquitoes, we are trying to determine optimal locations to release them in order to ensure the invasion of the environment by *Wolbachia*-infected mosquitoes. If we denote u the release function, then the above model is modified into

$$\begin{cases} \dfrac{\partial p(t,x)}{\partial t} - D\Delta p(t,x) = f(p(t,x)) + u(t,x)g(p(t,x)), & t \in (0,T), \quad x \in \Omega, \\ \partial_\nu p(t,x) = 0, & t \in (0,T),\ x \in \partial\Omega, \\ p(0,x) = 0, & x \in \Omega, \end{cases} \tag{9.1}$$

where Ω is an open bounded connected subset of $\mathbb{R}^d$ with a regular boundary $\partial\Omega$. The function g is positive and vanishes when $p = 1$. The derivation of (9.1) will be detailed in Section 2.

Let us summarize the main assumptions on f and g we will use in the sequel.

$$\begin{cases} f \text{ is } C^2 \text{ and of bistable type.} \\ \text{Denoting by } \theta \text{ the only root of } f \text{ in } (0,1), \text{ we assume that } f''(\cdot) > 0 \text{ on } (0,\theta). \\ g \text{ is nonnegative, decreasing on } [0,1]. \text{ Moreover, } g(1) = 0. \end{cases} \tag{$\mathcal{H}_{f,g}$}$$

A first study was carried out in [3], giving rise to the first very simple numerical experiments. In the present article, we seek to complete the results of this study, by analyzing qualitatively the solutions and by proposing adapted numerical strategies. Let us mention that a problem of the same nature has been investigated in [72], mainly from a numerical point of view. Authors characterize optimal vaccination strategies to minimizes the costs associated with infections by the Zika virus and vaccines in the state of Rio Grande do Norte in Brazil.

In [74], an optimal control problem close to the one investigated hereafter is tackled. The authors consider a population whose evolution is driven by a reaction-diffusion equation and look at determining initial data submitted to L^1 and L^∞ constraints, maximizing the total size of the population. In our article, we choose to deal with a least square criterion instead of the average criterion considered in [74] (and more recently in [68]). For reasons that will appear later, constant solutions are natural candidate to solve the considered optimal control problem. We show, as in [74], that for certain families of parameters, the constant functions are local minimizers of the optimal control problem. On the other hand, we complete this first analysis and also manage to show for our model, that these same functions can be or not be global minimizers depending on the considered range of parameters. These results are also illustrated numerically. Finally, it is worth mentioning that exact controllability issues for similar reaction-diffusion systems have been investigated in [64, 69].

The outline of the paper is the following. In Section 2, we present the derivation of system (9.1) and present the optimal control problem we are looking at. Section 3 contains the main mathematical results of this paper. Their proofs are given in Section 4. More precisely, the rigorous derivation of system (9.1) is explained in Section 4.1 and Section 4.2 is devoted to the mathematical study of the optimal control problem. Finally, numerical illustrations with the description of the numerical algorithm are provided in Section 5.

2 Modelling

In the whole article, we will consider a given bounded connected open domain Ω of $\mathbb{R}^d$ assumed to have a Lipschitz boundary. Let $T > 0$ denote a fixed horizon of time.

2.1 Model with two compartments

In order to justify the introduced model on the proportion of *Wolbachia*-infected mosquitoes, we first explain how to derive it. Let us denote n_{in} the density of infected mosquitoes and n_{un} the density of uninfected mosquitoes. The dynamics of these quantities is governed by the reaction-diffusion system

$$\begin{cases} (\partial_t - D\Delta)n_{in} = (1-s_f)\dfrac{F_{un}}{\varepsilon}n_{in}\left(1-\dfrac{N}{K}\right) - \delta d_{un}n_{in} + u, & \text{in } (0,T)\times\Omega \quad (9.2a) \\ (\partial_t - D\Delta)n_{un} = \dfrac{F_{un}}{\varepsilon}n_{un}\left(1-s_h\dfrac{n_{in}}{N}\right)\left(1-\dfrac{N}{K}\right) - d_{un}n_{un}, & \text{in } (0,T)\times\Omega \quad (9.2b) \\ N = n_{in} + n_{un} & \text{in } (0,T)\times\Omega \\ \partial_\nu n_{in} = 0, \quad \partial_\nu n_{un} = 0, & \text{on } (0,T)\times\partial\Omega \end{cases}$$

complemented by initial conditions $n_{in}(t=0,x) = n_{in}^{\text{init}}(x) \geqslant 0$, $n_{un}(t=0,x) = n_{un}^{\text{init}}(x) > 0$, where the following notations are used:

- u: instantaneous releases of *Wolbachia* infected mosquitoes. It is on this control that we will act upon. At this step, we do not make the admissible space of controls precise, this will be done in what follows;
- d_{un}, $d_{in} = \delta d_{un}$ with $\delta > 1$: death rates, respectively for uninfected and infected mosquitoes. We assume that $d_{in} > d_{un}$ since *Wolbachia* decreases lifespan;
- F_{un}, $F_{in} = (1-s_f)F_{un}$: net fecundity rates, respectively for uninfected and infected mosquitoes. We assume that $F_{in} < F_{un}$ since *Wolbachia* reduces fecundity;
- ε : parameter without dimension quantifying the fecundity, we assume $\varepsilon \ll 1$ meaning that the fecundity is considered to be large;
- $s_h \in (0,1)$: cytoplasmic incompatibility parameter (fraction of uninfected females' eggs fertilized by infected males which will not hatch). Formally, a proportion $1-s_h$ of uninfected female's eggs fertilized by infected males actually hatch. Cytoplasmic incompatibility is perfect when $s_h = 1$;
- K: carrying capacity;
- D: dispersal coefficient.

All the constants above are assumed to be positive. Existence and uniqueness of solutions for such reaction-diffusion system is by now well-known see e.g. [42, 78]. The equations driving the dynamics of n_{in} and n_{un} are bistable and monostable reaction-diffusion equations, respectively. Note that in the reaction term of the second equation, the term $-\frac{n_{in}}{n_{in}+n_{un}}$ stands for the vertical transition of the disease whereas the coefficient s_h models that this vertical transmission may or not be perfect because of the cytoplasmic incompatibility.

In accordance with [4], we will assume moreover that the relation

$$s_f + \delta - 1 < \delta s_h \tag{9.3}$$

holds true. It is notable that such a parameters choice is relevant since for *Wolbachia*-infected *Aedes* mosquitoes, and more precisely in the case of wMel strain, CI is almost perfect in these species-strain combination (see [40]) meaning that s_h is close to 1. Furthermore, such mosquitoes typically have a slightly reduced fecundity. In that particular case, one has $s_f \simeq 0.1$, $\delta \simeq 1.1$ and $s_h \simeq 0.9$ so that (9.3) holds true.

To model optimal strategies with an adapted optimal control problem, it is convenient to introduce the *Wolbachia*-infected equilibrium $(n^*_{in,W}, 0)$ for the uncontrolled system, defined by

$$(n^*_{in,W}, 0) := \left(K(1 - \frac{\varepsilon \delta d_{un}}{F_{un}(1 - s_f)}), 0 \right), \tag{9.4}$$

that is $(n^*_{in,W}, 0)$ is a stationary solution of (9.2a)–(9.2b). A possible approach hence consists in looking for controls steering the system as close as possible to the target state $(n^*_{in,W}, 0)$. In some sense, it stands for the research of a control strategy ensuring the persistence of infected mosquitoes at the time horizon T.

This leads to define the least squares functional J_T given by

$$J_T(u) = \frac{1}{2} \int_\Omega n_{un}(T,x)^2 \, dx + \frac{1}{2} \int_\Omega (n^*_{in,W} - n_{in}(T,x))_+{}^2 \, dx, \tag{9.5}$$

where (n_{in}, n_{un}) denotes the unique solution to the reaction-diffusion system (9.2a). Here, we use the notation $x_+ = \max\{x, 0\}$. Observe that the presence of this maximum in the definition of J_T does not induce non-differentiability since the mapping $x \mapsto x_+^2$ from $\mathbb{R}$ to $\mathbb{R}$ is C^1.

2.2 Reduction for large fecundity

When the fecundity is large compared to other parameters, it is relevant to consider the asymptotics $\varepsilon \to 0$, which allows us to reduce system (9.2a)–(9.2b). This reduction is inspired by [4] where the authors consider a differential system. We first explain formally how to reduce this system and state the main result, the rigorous approach is postponed to Section 4.1. Since n_{in} and n_{un} will depend on ε, we use the notation n^ε_{in} and n^ε_{un}.

Formal reasoning. We investigate formally the limit as $\varepsilon \to 0$ in the (9.2a)–(9.2b). From (9.2a)–(9.2b), we expect that $n^\varepsilon_{in} + n^\varepsilon_{un} = K + O(\varepsilon)$. Then, we introduce the variables

$$n^\varepsilon = \frac{1}{\varepsilon} \left(1 - \frac{n^\varepsilon_{in} + n^\varepsilon_{un}}{K} \right), \qquad p^\varepsilon = \frac{n^\varepsilon_{in}}{n^\varepsilon_{in} + n^\varepsilon_{un}},$$

where p^ε is the proportion of infected mosquitoes in the population. Consider a sequence $(u^\varepsilon)_{\varepsilon>0}$ of controls. From straightforward computations from (9.2a)–(9.2b), we deduce

$$\partial_t n^\varepsilon - D\Delta n^\varepsilon = -\frac{1 - \varepsilon n^\varepsilon}{\varepsilon} (F_{un} n^\varepsilon (s_h (p^\varepsilon)^2 - (s_f + s_h) p^\varepsilon + 1) - d_{un}((\delta - 1)p^\varepsilon + 1)) - \frac{u^\varepsilon}{\varepsilon K}, \tag{9.6}$$

$$\begin{aligned}\partial_t p^\varepsilon - D\Delta p^\varepsilon + \frac{2\varepsilon D}{1-\varepsilon n^\varepsilon}\nabla p^\varepsilon \cdot \nabla n^\varepsilon \\ = p^\varepsilon(1-p^\varepsilon)(F_{un} n^\varepsilon(s_h p^\varepsilon - s_f) + (1-\delta)d_{un}) + \frac{u^\varepsilon(1-p^\varepsilon)}{K(1-\varepsilon n^\varepsilon)}.\end{aligned} \tag{9.7}$$

Letting formally ε going to 0, assuming that $(n^\varepsilon, p^\varepsilon, u^\varepsilon)$ converges to (n^0, p^0, u^0), we deduce from (9.6) that the limit should satisfy the relation

$$n^0 = h(p^0, u^0) := \frac{d_{un}((\delta-1)p^0+1) - u^0/K}{F_{un}(s_h(p^0)^2 - (s_f+s_h)p^0+1)}. \tag{9.8}$$

Then, passing into the limit in (9.7), we deduce

$$\partial_t p^0 - D\Delta p^0 = p^0(1-p^0)(F_{un} n^0(s_h p^0 - s_f) + (1-\delta)d_{un}) + \frac{u^0(1-p^0)}{K}.$$

Injecting (9.8) into this latter equation, we obtain the scalar reaction-diffusion equation for the fraction of infected mosquitoes

$$\begin{cases} \partial_t p^0 - D\Delta p^0 = f(p^0) + u^0 g(p^0) & \text{in } (0,T)\times\Omega \\ \partial_n p^0 = 0 & \text{on } (0,T)\times\partial\Omega \end{cases} \tag{9.9}$$

with

$$f(p) = \frac{\delta d_{un} s_h p(1-p)(p-\theta)}{s_h p^2 - (s_f+s_h)p+1}, \qquad g(p) = \frac{(1-p)(1-s_h p)}{K(s_h p^2 - (s_f+s_h)p+1)}, \tag{9.10}$$

where we use the notation $\theta = \frac{s_f+\delta-1}{\delta s_h}$. Under the assumption (9.3) on the coefficients, we have $0<\theta<1$. Hence equation (9.9) for $u^0=0$ is a bistable reaction-diffusion equation.

Remark 2.1. *We claim that if the coefficients s_f and s_h satisfy*

$$(s_f+s_h)^2 < 4s_h \tag{9.11}$$

and

$$\delta(s_f+s_h-2) - s_f + 1 < 0, \tag{9.12}$$

then the particular functions f and g given by (9.10) satisfy assumption ($\mathcal{H}_{f,g}$).
Let us show it. Assuming that (9.11) holds true, we infer that $h(p) := s_h p^2 - (s_f+s_h)p+1 > 0$ for all $p\in\mathbb{R}$, which implies that f is C^2 and bistable. Straightforward computations yield

$$f''(p) = \frac{\delta d_{un} s_h \psi(p)}{2(1-s_f)h(p)^3},$$

where

$$\psi(p) = p^3 s_f(1-1/\delta) + p^3 s_h - 3p^2 + \frac{1}{s_h} + \frac{3p^2}{\delta} - \frac{3p}{\delta} + \frac{s_f}{\delta s_h} + \frac{1}{\delta} - \frac{1}{\delta s_h}.$$

Hence, f'' and ψ share the same sign on $(0,1)$. Since $\delta>1$, $s_f\in(0,1)$ and $s_h\in(0,1)$, one has $\psi^{(3)} = 6(\delta s_f + \delta s_h - s_f)/\delta \geqslant 0$, i.e. ψ'' is increasing. We deduce that ψ' is decreasing on a interval $(0,a)$ and increasing on $(a,1)$ with $a\in[0,1]$. Furthermore, one has $\psi'(0) = -3/\delta < 0$ and $\psi'(1) < 0$ if we assume (9.12). In that case, ψ is decreasing on $(0,1)$. In addition $\psi(0) = (\delta+s_f+s_h-1)/(\delta s_h) > 0$, thus, under (9.11)-(9.12), we have $f''>0$ on $(0,\theta)$. We remark that g' is negative on $(-1/\sqrt{s_h}, 1/\sqrt{s_h})$. Since $s_h\in(0,1)$, we deduce that g decreases on $(0,1)$. Moreover $g(1)=0$ and therefore, g is positive on $(0,1)$.

We introduce the notation F for the antiderivative of f,

$$F(p) = \int_0^p f(q)\,dq.$$

In what follows, we will assume:

$$\exists \theta_c \in (\theta, 1) \quad | \quad F(\theta_c) = 0. \tag{9.13}$$

This assumption is necessary to guarantee that invasion of the infected population may occur in space by local release. We will check that this assumption is satisfied for the particular choice of parameters we will consider for the numerical experiments in Remark 5.1.

We consider system (9.2a)–(9.2b) with Neumann boundary conditions to model that the boundary acts as a barrier, and initial conditions satisfying

$$n_{in}^{\text{init},\varepsilon} \in L^\infty(\Omega), \quad 0 \leqslant n_{in}^{\text{init},\varepsilon}, \quad n_{un}^{\text{init},\varepsilon} \in L^\infty(\Omega), \quad 0 < n_{un}^{\text{init},\varepsilon}. \tag{9.14}$$

We assume also that the initial conditions are well-prepared, i.e.

$$n_{in}^{\text{init},\varepsilon} + n_{un}^{\text{init},\varepsilon} = K + \varepsilon K_0^\varepsilon, \qquad \text{with } \|K_0^\varepsilon\|_\infty \leqslant C. \tag{9.15}$$

A typical example of initial conditions is when the system is as the *Wolbachia*-free equilibrium for which $n_{in}^{\text{init},\varepsilon} = 0$ and $n_{un}^{\text{init},\varepsilon} = K(1 - \frac{\varepsilon d_{un}}{F_{un}})$. In this case, assumption (9.15) is obviously satisfied.

Convergence result. Following the ideas in [89], where a similar asymptotic limit is performed, we derive an asymptotic model on the proportion of infected mosquitoes, as the fecundity rates tend to $+\infty$.

Theorem 2.1. *Under the assumptions (9.14)–(9.15) on the initial data, let us assume moreover that the sequence (u^ε) converges towards u^0, weakly star in $L^\infty((0,T)\times\Omega)$ as $\varepsilon \searrow 0$. Then, up to extraction of subsequences, the solution $(n^\varepsilon, p^\varepsilon)$ of (9.6)–(9.7) converges towards (n^0, p^0) as $\varepsilon \to 0$, with $n^0 \in L^\infty((0,T)\times\Omega)$, $p^0 \in L^2(0,T;H^1(\Omega))$, and satisfying (9.8) almost everywhere and (9.9)–(9.10) in the weak sense. More precisely, we have*

$$p^\varepsilon \to p^0 \text{ strongly in } L^2((0,T)\times\Omega), \qquad n^\varepsilon \rightharpoonup n^0 \text{weakly-star in } L^\infty((0,T)\times\Omega),$$

where p^0 is solution to (9.9)

The proof of this theorem is postponed to Section 4.1.

Let us now define the least squares functional J_T^ε given by

$$J_T^\varepsilon(u) = \frac{1}{2}\int_\Omega n_{un}^\varepsilon(T,x)^2\,dx + \frac{1}{2}\int_\Omega \left(n_{in,W}^* - n_{in}^\varepsilon(T,x)\right)_+^2 dx, \tag{9.16}$$

where $(n_{in}^\varepsilon, n_{un}^\varepsilon)$ have been introduced at the very beginning of Section 2.2 and $n_{in,W}^*$ in (9.4). As a corollary of the convergence result above, let us make the asymptotic behavior of the functional J_T^ε precise.

Corollary 2.2. *Under the assumptions (9.14)–(9.15) on the initial data, let (u^ε) be a sequence converging towards u^0, weakly star in $L^\infty((0,T)\times\Omega)$ as $\varepsilon \searrow 0$. Then, $p^\varepsilon(T,\cdot)$ converges towards $p^0(T,\cdot)$ strongly in $L^2(\Omega)$, and moreover, $J_T^\varepsilon(u_\varepsilon)$ converges towards $J_T^0(u^0)$ defined by*

$$J_T^0(u) = K^2 \int_\Omega (1 - p^0(T,x))^2\, dx, \tag{9.17}$$

where p denotes the solution of (9.9), as $\varepsilon \searrow 0$.

The proof of this result is postponed to Section 4.1.

In what follows, we will rather deal with the proportion p^0 to model optimal releases strategies. The following section is dedicated to modeling issues about the optimal control problem we will deal with.

2.3 Toward an optimal control problem

In this section, we will introduce an optimal control problem modeling optimal mosquito releases. For this purpose, we assume all fecundity rates large, which legitimates the use of the asymptotic model (9.9)–(9.10) introduced in Section 2.2.. We will focus on time-pulsed releases, which will lead us to further simplify the problem.

In order not to cumulate all the difficulties related to the search for release distributions in time and space, we will suppose that one release, which is an impulse in time[2], is done at the beginning of the experiment, i.e. $u(t,x)$ will be assimilated to a particular approximation of a Dirac impulse in time, namely $u_0(x)\delta_{\{t=0\}}$.
More precisely, we will consider as choice of release term, the function

$$u^0(t,x) = \frac{1}{\eta}\mathbb{1}_{[0,\eta]}(t)u_0(x),$$

where $u_0 \in L^\infty(\Omega)$ will be given. Making the change of variable $t = \tau\eta$, and introducing $\tilde{p}$ given by $\tilde{p}(\tau,x) = p^0(t,x)$, one gets from system (9.9) that $\tilde{p}$ solves

$$\frac{\partial \tilde{p}}{\partial \tau} - \eta D\Delta\tilde{p} = \eta f(\tilde{p}) + u_0 g(\tilde{p}), \qquad \tau \in [0,1],\ x \in \Omega.$$

We now provide a purely formal argument to justify the optimal control problem we will deal with.

Letting formally η go to 0 and denoting, with a slight abuse of notation, still by $\tilde{p}$ the formal limit of the system above yields

$$\frac{\partial \tilde{p}}{\partial \tau}(\tau,x) = u_0(x)g(\tilde{p}(\tau,x)), \qquad \tau \in [0,1],\ x \in \Omega. \tag{9.18}$$

Let us denote G the anti-derivative of $1/g$ vanishing at 0, namely

$$G(p) = \int_0^p \frac{dq}{g(q)}.$$

[2]We consider Dirac measures since at the time-level of the study (namely, some generations), the release can be considered as instantaneous.

Then, by a direct integration of (9.18) on $[0,1]$, we obtain

$$G(\tilde{p}(1,x)) = G(\tilde{p}(0,x)) + u_0(x), \qquad x \in \Omega.$$

Hence we arrive at the system

$$\begin{cases} \dfrac{\partial p}{\partial t} - D\Delta p = f(p), & t \in (0,T), \quad x \in \Omega, \\ \quad \partial_\nu p(t,x) = 0, & x \in \partial\Omega, \\ \quad\ p(0^+,\cdot) = G^{-1}(u_0(\cdot)), \end{cases} \tag{9.19}$$

where f and g are given by (9.10).

According to $(\mathcal{H}_{f,g})$, $G(0) = 0$, $G(1^-) = +\infty$, G is continuous in $[0,1)$ and strictly increasing, $G^{-1}(u_0)$ is well defined and in $[0,1)$ for positive u_0. Moreover 0 and 1 are subsolution and uppersolution to (9.19), hence thanks to a standard comparison argument for parabolic systems, the solution p to System (9.19) satisfies $0 \leqslant p(t,x) < 1$ for a.e. $t \in [0,T]$ and $x \in \Omega$ (see e.g. [21]).

To take into account biological constraints on the release procedure, we will moreover assume that the release function is such that:

- the local release of mosquitoes is bounded : $0 \leqslant u_0 \leqslant M$ a.e. in Ω with $M > 0$;
- the total number of used mosquitoes is bounded (production limitation), reading

$$\int_\Omega u_0(x)\,dx \leqslant C,$$

  with $C \in (0, MT)$. Note that it is relevant to choose the parameter C strictly lower than MT. In the converse case, it would mean that the choice $u_0(\cdot) = M$ is admissible, so that the local maximal number of mosquitoes can be released (almost) everywhere in Ω. Since producing infected mosquitoes has an important cost, it is reasonable from a biological point a view to assume that such a release is not possible.

This leads to introduce the admissible set $\mathcal{V}_{C,M}$ given by

$$\mathcal{V}_{C,M} = \left\{ u_0 \in L^\infty(\Omega), 0 \leqslant u_0 \leqslant M \text{ a.e. in } \Omega, \ \int_\Omega u_0(x)\,dx \leqslant C \right\}.$$

The goal is to be as near as possible to the equilibrium $p = 1$ at time T. Let us denote (with a slight abuse of notation) by J_T, the least squares functional defined by

$$J_T(u_0) = \frac{1}{2}\int_\Omega (1 - p(T,x))^2\,dx.$$

Observe that coincides, up to a positive multiplicative constant, with the asymptotic functional J_T^0 given by (9.17). The optimization problem thus reads

$$\boxed{\inf_{u_0 \in \mathcal{V}_{C,M}} J_T(u_0)}, \tag{$\mathcal{P}_{\text{reduced}}$}$$

where p is the solution of (9.19).

From now on and without loss of generality, we will assume in what follows that the diffusion coefficient D is equal to 1.

3 Main results

Constant solutions are natural candidates to solve Problem ($\mathcal{P}_{\text{reduced}}$). Indeed, it has been observed in [3, Theorem 2.1] that in the very simple case where $f(\cdot) = 0$ and $G : x \mapsto x$, Problem ($\mathcal{P}_{\text{reduced}}$) has a unique solution u_0, which is constant and equal to $\min\left(1, M, \frac{C}{|\Omega|}\right)$. Furthermore, as stated in the following result, constant solutions equal to M are optimal for a given range of the parameters. We show moreover that, outside of this range, constant functions remain critical points and show that they are still local minimizers whenever C is small enough. We also comment on the sharpness of this result by highlighting that for certain parameters, constant functions may not be global minimizers for Problem ($\mathcal{P}_{\text{reduced}}$).

According to Corollary 4.7, it is enough to concentrate on the constant function equal to $C/|\Omega|$.

Theorem 3.1. *Let us assume that f and g satisfy ($\mathcal{H}_{f,g}$). Problem ($\mathcal{P}_{\text{reduced}}$) has a solution.*

(i) For every $M \in (0, C/|\Omega|]$, the constant function $\overline{u}_M$ equal to M is the unique solution to Problem ($\mathcal{P}_{\text{reduced}}$).

(ii) Let us assume that $M|\Omega| > C$. The constant function $\overline{u}(\cdot) = C/|\Omega|$ is a critical point for Problem ($\mathcal{P}_{\text{reduced}}$) (meaning that it satisfies the first order optimality conditions stated in Proposition 4.6).

Furthermore, if $C \leqslant |\Omega| G(\theta)$, there exists $K_T > 0$ such that for every $h \in L^2(\Omega)$, the second order differential of J_T at $\overline{u}$ satisfies

$$d^2 J_T(\overline{u})(h,h) \geqslant K_T \|h\|^2_{L^2(\Omega)}$$

and it follows that the function $\overline{u}$ is a local minimizer for Problem ($\mathcal{P}_{\text{reduced}}$).

Let us comment on the sharpness of Theorem 3.1. As will be emphasized hereafter and in Section 5, we do not expect that $\overline{u}$ solves Problem ($\mathcal{P}_{\text{reduced}}$) for all values of $C \in (|\Omega| G(\theta), M|\Omega|)$. In some case, this will be confirmed numerically, by using $\overline{u}$ as starting point of optimization algorithms and obtain at convergence a nonconstant minimizer $\tilde{u}$ such that $J_T(\tilde{u}) < J_T(\overline{u})$.

Actually, even in the case $C < \min\{G(\theta), M\}|\Omega|$, where we know from Theorem 3.1 that the constant solution $\overline{u}$ is a local minimizer, under some conditions on $|\Omega|$ and C, we may construct non constant initial date u_0 such that $J_T(u_0) < J_T(\overline{u})$, as stated below in Proposition 4.11.

Recalling that $\theta_c \in (\theta, 1)$ is defined by $\int_0^{\theta_c} f(p)\, dp = 0$. We assume

$$G(\theta_c) < M \quad \text{and} \quad C < |\Omega| G(\theta). \tag{9.20}$$

The following result shows that under some conditions on Ω, M and C, the constant function $\overline{u}$ is not a global minimum of the optimization problem ($\mathcal{P}_{\text{reduced}}$).

Proposition 3.2. *Let us assume (9.20) and that:*

- *C is large enough;*
- *the inradius[3] of Ω is large enough;*

[3] In other words, the radius of the largest ball inscribed in Ω.

- *T is large enough.*

Then the constant solution $\overline{u}$ is not a global minimum for Problem ($\mathcal{P}_{\text{reduced}}$).

A proof of this result is provided in Section 4.2.

Remark 3.1. *The conditions stated in Proposition 3.2 are not sharp; the obtention of necessary and sufficient condition for constant solution to be a global minimizers seems to be intricate and we let it open. A related problem concerns the issue of finding sufficient and necessary conditions guaranteeing invasion in a bistable reaction-diffusion system that is, up to our knowledge still open, and we refer to [74] for partial answers in this direction.*

Remark 3.2. *It is notable that, for the sets of parameters from [46] below, the functions f and g satisfy ($\mathcal{H}_{f,g}$). Indeed, the function f'' vanishes once on $(0,1)$ and its root z satisfies $\theta < z \simeq 0.466$, while $\theta_c \simeq 0.582$ (see Figure 9.1).*

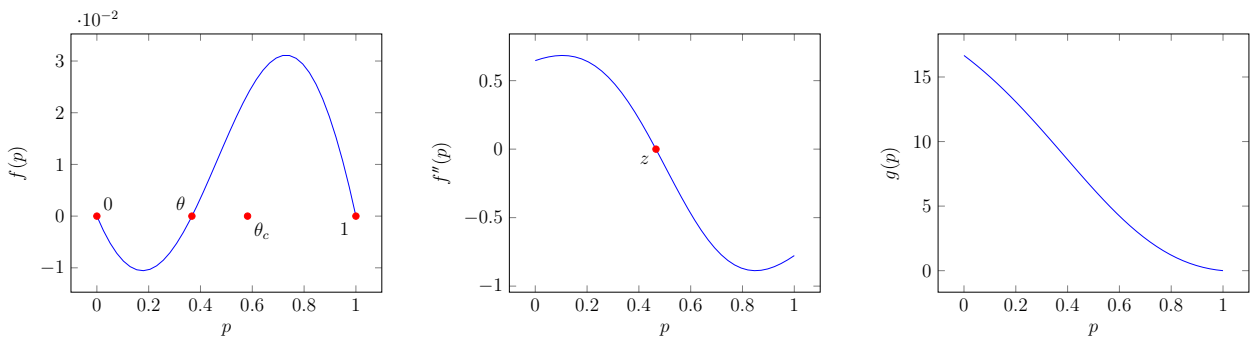

Figure 9.1: (from left to right) Graphs of the function f, its second order derivative f'' and the function g by using the data from [46] (see Table 9.1).

4 Proofs

This section will be devoted to prove Theorem 2.1 and 3.1.

4.1 Model reduction

In this section, we will give a proof of Theorem 2.1, allowing us to reduce the system (9.2a)–(9.2b) to a scalar reaction-diffusion equation for the proportion as the parameter ε goes to 0. It is inspired by [4] in which the authors use a model composed by two differential equations.

Uniform a priori estimates

We first establish some uniform bounds with respect to $\varepsilon > 0$.

Lemma 4.1. *Assume the assumptions of Theorem 2.1 hold. Let u^ε be given in $\mathcal{V}_{C,M}$, $\varepsilon \in (0,1)$ and $(p^\varepsilon, n^\varepsilon)$ be the unique solution of (9.6)–(9.7). Then,*

$$n^\varepsilon \text{ is uniformly bounded in } L^\infty([0,T]\times\Omega), \text{ and } 0 \leqslant p^\varepsilon \leqslant 1 \text{ on } [0,T]\times\Omega.$$

Proof. By nonnegativity of n_{in}^{init} and n_{un}^{init}, it is standard to deduce the nonnegativity of n_{in}^{ε} and n_{un}^{ε} (Indeed 0 is a subsolution for (9.2a) and for (9.2b), see e.g. [21]). Moreover, since $\mathbb{R}_+$ is invariant for the equation of n_{un}^{ε} and n_{un}^{init} is non identically equal to zero, we deduce that $n_{un}^{\varepsilon} > 0$ on $\Omega \times (0,T]$ (see e.g. [95, th. 2]). Therefore, p^{ε} is well-defined on $[0,T] \times \Omega$ and satisfies by definition $0 \leqslant p^{\varepsilon} \leqslant 1$ on $[0,T] \times \Omega$.

Consider the function h defined in (9.8). We remark that the denominator is positive. Let $\tilde{K} := \max\{\max_{p\in[0,1]} h(p,0), \|n^{\text{init},\varepsilon}\|_\infty\}$. Let ε_0 be such that $\tilde{K} \leqslant \frac{1}{\varepsilon_0}$ ($\Leftrightarrow 1 - \varepsilon_0 \tilde{K} \geqslant 0$), then, thanks to this choice, we have for $0 < \varepsilon \leqslant \varepsilon_0$

$$\begin{aligned}
&-\frac{1-\varepsilon\tilde{K}}{\varepsilon}\Big(F_{un}\tilde{K}(s_h(p^{\varepsilon})^2 - (s_f+s_h)p^{\varepsilon}+1) - d_{un}((\delta-1)p^{\varepsilon}+1)\Big) - \frac{u^{\varepsilon}}{\varepsilon K}\\
&\leqslant -\frac{1-\varepsilon\tilde{K}}{\varepsilon}F_{un}(s_h(p^{\varepsilon})^2 - (s_f+s_h)p^{\varepsilon}+1)(\tilde{K} - h(p,0))\\
&\leqslant 0 = \partial_t\tilde{K} - D\Delta\tilde{K}.
\end{aligned}$$

Hence, we have that $\tilde{K}$ is a supersolution for (9.6) for any $0 < \varepsilon \leqslant \varepsilon_0$. Then $n^{\varepsilon} \leqslant \tilde{K}$ for any $0 < \varepsilon \leqslant \varepsilon_0$.

By the same token, we have that the negative constant $\min\{-\|n^{\text{init},\varepsilon}\|_\infty, \min_{p\in[0,1]} h(p,M)\}$ is a subsolution for (9.6). Thus n^{ε} is uniformly bounded from below. We deduce the uniform bound of n^{ε} in $L^\infty([0,T]\times\Omega)$. □

Lemma 4.2. *Under above assumptions, for $\varepsilon > 0$ small enough, we have the uniform estimate*

$$\int_0^T \int_\Omega |\nabla p^{\varepsilon}|^2 \, dx \leqslant \overline{C} \tag{9.21}$$

and

$$\varepsilon D \int_0^T \int_\Omega |\nabla n^{\varepsilon}|^2 \, dxdt \leqslant C_0, \tag{9.22}$$

for some nonnegative constants $\overline{C}$ and C_0.

Proof. On the one hand, multiplying equation (9.6) by $\varepsilon n^{\varepsilon}$ and integrating on Ω, we get

$$\begin{aligned}
&\varepsilon\frac{d}{dt}\int_\Omega |n^{\varepsilon}|^2 \, dx + \varepsilon D \int_\Omega |\nabla n^{\varepsilon}|^2 \, dx\\
&\quad = -\int_\Omega (1-\varepsilon n^{\varepsilon}) n^{\varepsilon}(F_{un} n^{\varepsilon}(s_h(p^{\varepsilon})^2 - (s_f+s_h)p^{\varepsilon}+1) - d_{un}((\delta-1)p^{\varepsilon}+1)) \, dx\\
&\qquad - \frac{1}{K}\int_\Omega u^{\varepsilon} n^{\varepsilon} \, dx.
\end{aligned}$$

Since from Lemma 4.1, we know that n^{ε} and p^{ε} are uniformly bounded in $L^\infty([0,T]\times\Omega)$, we deduce (9.22) after an integration in time.

On the other hand, we fix $\varepsilon_0 > 0$ small enough such that, for all $\varepsilon \leqslant \varepsilon_0$, we have $|n^{\varepsilon}| \leqslant C_1 < \frac{1}{\varepsilon}$ on $[0,T]\times\Omega$ for some constant $C_1 > 0$ (which is always possible thanks to Lemma 4.1).

Then, we multiply by p^ε the equation satisfied by p^ε (9.7) and integrate over Ω, we deduce

$$\frac{1}{2}\frac{d}{dt}\int_\Omega (p^\varepsilon)^2\,dx + D\int_\Omega |\nabla p^\varepsilon|^2\,dx + 2\varepsilon D\int_\Omega \frac{p^\varepsilon}{1-\varepsilon n^\varepsilon}\nabla p^\varepsilon\cdot\nabla n^\varepsilon\,dx \leqslant C_3 + \frac{1}{K(1-\varepsilon_0 C_1)}\int_\Omega u^\varepsilon(t,x)\,dx$$

for some nonnegative constant C_3. Then, using a Cauchy-Schwarz inequality, we get

$$\frac{1}{2}\frac{d}{dt}\int_\Omega (p^\varepsilon)^2\,dx + D\int_\Omega |\nabla p^\varepsilon|^2\,dx \leqslant C_2 + \frac{2\varepsilon D}{1-\varepsilon C_1}\left(\int_\Omega |\nabla p^\varepsilon|^2\,dx\right)^{1/2}\left(\int_\Omega |\nabla n^\varepsilon|^2\,dx\right)^{1/2} + \frac{1}{K(1-\varepsilon C_1)}\int_\Omega u^\varepsilon(t,x)\,dx.$$

From (9.22) and the well-known inequality $2ab \leqslant a^2+b^2$, we deduce after an integration in time

$$\frac{1}{2}\int_\Omega (p^\varepsilon)^2\,dx + D\left(1-\frac{\varepsilon}{1-\varepsilon C_0}\right)\int_0^T\int_\Omega |\nabla p^\varepsilon|^2\,dxdt \leqslant C_2 T + \frac{DC_0^{1/2}}{1-\varepsilon C_1} + \frac{MT|\Omega|}{K(1-\varepsilon C_1)},$$

where we recall that $u^\varepsilon \in \mathcal{V}_{C,M}$ and $p\in[0,1]$. Taking ε small enough, we get the desired estimate. □

Compactness result and proof of Theorem 2.1

We first recall the following compactness result (see [83]).

Lemma 4.3 (Aubin-Lions)**.** *Let $T>0, q\in(1,\infty)$, $(\psi_n)_n$ a bounded sequence in $L^q(0,T;H)$, where H is a Banach space. If ψ_n is bounded in $L^q(0,T;V)$ and V compactly embeds in H, and if $(\partial_t\psi_n)_n$ is bounded in $L^q(0,T;V')$ uniformly with respect to n, then $(\psi_n)_n$ is relatively compact in $L^q(0,T;H)$.*

Proof of Theorem 2.1 We split the proof of the Theorem into several steps.

Step 1. Compactness. We use Lemma 4.3 with $q=2$, $H=L^2(\Omega)$, $V=H^1(\Omega)\cap L^\infty(\Omega)$. Then, the sequence (p^ε) is clearly bounded in $L^2(0,T;V)$ from Lemma 4.1 and 4.2. The compact embedding of V in H is well-known from the Rellich-Kondrachov Theorem. We are left to verify the bound on the time derivative: let $\phi\in V$, we denote $\langle\cdot,\cdot\rangle := \langle\cdot,\cdot\rangle_{V',V}$ the duality bracket. From equation (9.7), we get

$$\int_0^T |\langle\partial_t p^\varepsilon(t),\phi\rangle|^2\,dt = \int_0^T \left|\langle D\Delta p^\varepsilon(t) - \frac{2\varepsilon D}{1-\varepsilon n^\varepsilon(t)}\nabla p^\varepsilon(t)\cdot\nabla n^\varepsilon(t) - \psi^\varepsilon(t),\phi\rangle\right|^2 dt,$$

where ψ^ε is the function defining the right hand side in equation (9.7), which is uniformly bounded in $L^1((0,T)\times\Omega)\cap L^\infty((0,T)\times\Omega)$ as a direct consequence of Lemma 4.1. Then,

$$\begin{aligned}\int_0^T |\langle\partial_t p^\varepsilon,\phi\rangle|^2\,dt \leqslant{}& C_0\|\nabla\phi\|^2_{L^2(\Omega)}\int_0^T \|\nabla p^\varepsilon\|^2_{L^2(\Omega)}\,dt \\ &+ C_1\varepsilon\|\phi\|^2_{L^\infty(\Omega)}\int_0^T \|\nabla p^\varepsilon\|^2_{L^2(\Omega)}\|\nabla n^\varepsilon\|^2_{L^2(\Omega)}\,dt \\ &+ C_2\|\phi\|^2_{L^2(\Omega)}\|\psi^\varepsilon\|^2_{L^\infty((0,T)\times\Omega)}.\end{aligned}$$

Hence, we get the required bound from Lemma 4.1 and 4.2 and estimate (9.22). We may apply Lemma 4.3 and deduce the relative strong compactness of (p_ε) in $L^2(0,T;L^2(\Omega))$.

Moreover, from the estimates in Lemma 4.1, we deduce the relative weak-star compactness of the sequence (n_ε) in $L^\infty((0,T)\times\Omega)$. Therefore, there exists $p^0 \in L^2(0,T;H^1(\Omega))$ and $n^0 \in L^\infty((0,T)\times\Omega)$ such that, up to extraction of subsequences, we have $p_\varepsilon \to \overline{p}$ strongly in $L^2((0,T)\times\Omega)$ and a.e., $\nabla p_\varepsilon \rightharpoonup \nabla\overline{p}$ weakly in $L^2((0,T)\times\Omega)$, and $n_\varepsilon \rightharpoonup \overline{n}$ in $L^\infty((0,T)\times\Omega)$ -weak$\star$.

Step 2. Passing to the limit. We now pass to the limit in the weak formulation of equations (9.6) and (9.7). From the weak formulation of (9.6), we deduce that for any test function $\phi \in C_c^\infty((0,T)\times\Omega)$, we have

$$\begin{aligned}
&\varepsilon\int_0^T\int_\Omega(-n^\varepsilon\partial_t\phi - Dn^\varepsilon\Delta\phi)\,dxdt\\
&= -\int_0^T\int_\Omega\left(F_{un}n^\varepsilon(s_h(p^\varepsilon)^2-(s_f+s_h)p^\varepsilon+1)-d_{un}((\delta-1)p^\varepsilon+1))-\frac{u^\varepsilon}{K}\right)\phi\,dxdt\\
&\quad+\varepsilon\int_0^T\int_\Omega n^\varepsilon(F_{un}n^\varepsilon(s_h(p^\varepsilon)^2-(s_f+s_h)p^\varepsilon+1)-d_{un}((\delta-1)p^\varepsilon+1))\phi\,dxdt.
\end{aligned}$$

From the L^∞-bound of Lemma 4.1, we deduce that the term of the left hand side and the last term of the right hand side converge to 0 as $\varepsilon\to 0$. For the first term of the right hand side, we may pass into the limit thanks to the weak convergence of n^ε, the strong convergence of p^ε, and the weak convergence of u^ε. We obtain, for any $\phi\in C_c^\infty((0,T)\times\Omega)$,

$$0 = -\int_0^T\int_\Omega\left(F_{un}n^0(s_h(p^0)^2-(s_f+s_h)p^0+1)-d_{un}((\delta-1)p^0+1))-\frac{u^0}{K}\right)\phi\,dxdt.$$

As a consequence (9.8) is verified almost everywhere.

We are left to pass into the limit in the weak formulation of (9.7). Let $\phi\in C_c^\infty([0,T]\times\overline{\Omega})$, we have

$$\begin{aligned}
&\int_0^T\int_\Omega(-p^\varepsilon\partial_t\phi + D\nabla p^\varepsilon\cdot\nabla\phi + \frac{2\varepsilon D\phi}{1-\varepsilon n^\varepsilon}\nabla p^\varepsilon\cdot\nabla n^\varepsilon)\,dxdt\\
&=\int_0^T\int_\Omega p^\varepsilon(1-p^\varepsilon)(F_{un}n^\varepsilon(s_hp^\varepsilon-s_f)+(1-\delta)d_{un})\phi\,dxdt+\int_0^T\int_\Omega\frac{u^\varepsilon(1-p^\varepsilon)}{K(1-\varepsilon n^\varepsilon)}\phi\,dxdt.
\end{aligned}\tag{9.23}$$

From the above convergence it is straightforward to pass into the limit into the first two terms of the left hand side. For the third term, we use estimate (9.22), and a Cauchy-Schwarz inequality to get

$$\int_0^T\int_\Omega\frac{2\varepsilon D\phi}{1-\varepsilon n^\varepsilon}\nabla p^\varepsilon\cdot\nabla n^\varepsilon\,dxdt \leqslant \frac{2\sqrt{\varepsilon}D\|\phi\|_{L^\infty}}{1-\varepsilon\|n^\varepsilon\|_{L^\infty}}\|\nabla p^\varepsilon\|_{L^2}\sqrt{C}\to 0,$$

as $\varepsilon\to 0$, thanks to Lemma 4.1 and 4.2.

We may pass into the limit for the first term of the right hand side of (9.23) since (p^ε) converges strongly and a.e., and (n^ε) converges weakly. Then, for the last term of the right hand side of (9.23), we verify

$$\begin{aligned}
&\left|\int_0^T\int_\Omega\left(\frac{u^\varepsilon(1-p^\varepsilon)}{K(1-\varepsilon n^\varepsilon)}-\frac{u^0(1-p^0)}{K}\right)\phi\,dxdt\right|\\
&=\left|\int_0^T\int_\Omega\left(\frac{(u^\varepsilon-u^0)(1-p^0)+u^\varepsilon(p^0-p^\varepsilon)+\varepsilon u^0(1-p^0)n^\varepsilon}{K(1-\varepsilon n^\varepsilon)}\right)\phi\,dxdt\right|\\
&\leqslant\left|\int_0^T\int_\Omega\left(\frac{(u^\varepsilon-u^0)(1-p^0)}{K(1-\varepsilon n^\varepsilon)}\right)\phi\,dxdt\right|+\frac{\|u^0\|_{L^\infty}\|\phi\|_{L^2}}{K(1-\varepsilon\|n^\varepsilon\|_{L^\infty})}\|p^0-p^\varepsilon\|_{L^2}\\
&\qquad+\varepsilon\frac{\|u^\varepsilon\|_{L^\infty}\|n^\varepsilon\|_{L^\infty}\|\phi\|_{L^1}}{K(1-\varepsilon\|n^\varepsilon\|_{L^\infty})}.
\end{aligned}\tag{9.24}$$

From the strong L^2 convergence of (p^ε) and the L^∞ bounds in Lemma 4.1, we deduce that the last two terms go to 0 as $\varepsilon\to 0$. For the first term, we write

$$\frac{(u^\varepsilon-u^0)(1-p^0)}{K(1-\varepsilon n^\varepsilon)}=\frac{(u^\varepsilon-u^0)(1-p^0)}{K}+\varepsilon\frac{(u^\varepsilon-u^0)(1-p^0)n^\varepsilon}{K(1-\varepsilon n^\varepsilon)}.$$

It is then straightforward to conclude the convergence towards 0 of the first term of the right hand side of (9.24).

Finally, passing into the limit $\varepsilon\to 0$ into (9.23), we obtain

$$\begin{aligned}
&\int_0^T\int_\Omega(-p^0\partial_t\phi+D\nabla p^0\cdot\nabla\phi)\,dxdt=\\
&\int_0^T\int_\Omega p^0(1-p^0)(F_{un}n^0(s_hp^0-s_f)+(1-\delta)d_{un})\phi\,dxdt+\int_0^T\int_\Omega\frac{u^0(1-p^0)}{K}\phi\,dxdt.
\end{aligned}$$

We conclude by using the fact that (n^0,p^0,u^0) verifies the relation (9.8).

Proof of Corollary (2.2)

First, observe that $p^\varepsilon(T,\cdot)$ converges weakly-star to $p^0(T,\cdot)$ in $L^2(\Omega)$. Indeed, the proof is standard. Consider the variational formulation (9.23) on p^ε where test functions ϕ are now chosen in $C^\infty([0,T]\times C_c^\infty(\overline{\Omega}))$ instead of $C_c^\infty([0,T]\times\overline{\Omega})$. The variational formulation (9.23) is then modified by the addition of

$$\int_\Omega(p^\varepsilon(T,\cdot)\phi(T,\cdot)-p^\varepsilon(0,\cdot)\phi(0,\cdot))\,dx$$

in the left-hand side term. Since $p^\varepsilon(T,\cdot)$ is bounded in $L^2(\Omega)$ it converges weakly to some limit in $L^2(\Omega)$ up to a subsequence. Passing to the limit in the variational formulation and using Theorem 2.1 allows us to identify the closure point of $p^\varepsilon(T,\cdot)$ as $p^0(T,\cdot)$. Finally, by uniqueness of the closure point, we infer that the whole sequence $p^\varepsilon(T,\cdot)$ converges to $p^0(T,\cdot)$. Since the $L^2(\Omega)$-norm is lower semicontinuous for the weak-topology, we get that

$$\liminf_{\varepsilon\to 0}\int_\Omega\left(p^\varepsilon(T,x)\right)^2dx\leqslant\int_\Omega\left(p^0(T,x)\right)^2dx.$$

Let us multiply the equation (9.7) by p^ε and then, integrate it on $(0,T)\times\Omega$, we obtain

$$\begin{aligned}\frac{1}{2}\int_\Omega \left(p^\varepsilon(T)\right)^2 dx = \frac{1}{2}\int_\Omega (p^{\text{init},\varepsilon})^2\,dx - D\int_0^T\int_\Omega |\nabla p^\varepsilon|^2\,dx + \int_0^T\int_\Omega \frac{u^\varepsilon p^\varepsilon(1-p^\varepsilon)}{K(1-\varepsilon n^\varepsilon)}\,dx \\ + \int_0^T\int_\Omega (p^\varepsilon)^2(1-p^\varepsilon)(F_u n^\varepsilon(s_h p^\varepsilon - s_f) + (1-\delta)d_u)\,dx \\ - 2\varepsilon D\int_0^T\int_\Omega \frac{p^\varepsilon}{1-\varepsilon n^\varepsilon}\nabla p^\varepsilon\cdot\nabla n^\varepsilon\,dx.\end{aligned} \tag{9.25}$$

By assumption on the initial data, we have the convergence of the first term of the right hand side. For the second term of the right hand side, the weak convergence in $L^2((0,T),H^1(\Omega))$ guarantee that this term is upper semi-continuous, then

$$\limsup_{\varepsilon\to 0}\left(-D\int_0^T\int_\Omega |\nabla p^\varepsilon|^2\,dx\right) \leqslant -D\int_0^T\int_\Omega |\nabla p^0|^2\,dx.$$

Due to the strong convergence in $L^2((0,T)\times\Omega)$ of the sequence $(p^\varepsilon)_\varepsilon$ and using also the uniform bound of the sequence $(n^\varepsilon)_\varepsilon$ (see Lemma 4.1), we deduce the convergence of the third term of the right hand side of (9.25). Using also the strong convergence of $(p^\varepsilon)_\varepsilon$ and the weak convergence of $(n^\varepsilon)_\varepsilon$, we get the convergence of the fourth term in the right hand side of (9.25). Finally, from a Cauchy-Schwarz inequality we have

$$2\varepsilon D\int_0^T\int_\Omega \frac{p^\varepsilon}{1-\varepsilon n^\varepsilon}\nabla p^\varepsilon\cdot\nabla n^\varepsilon\,dx \leqslant C\varepsilon^{\frac{1}{2}}\left(\int_0^T\int_\Omega |\nabla p^\varepsilon|^2\,dxdt\right)^{1/2}\left(\varepsilon\int_0^T\int_\Omega |\nabla n^\varepsilon|^2\,dxdt\right)^{1/2}.$$

Thanks to the estimates in Lemma 4.2, we get that this latter term goes to 0 as $\varepsilon\to 0$. Finally, we have proved that

$$\limsup_{\varepsilon\to 0}\int_\Omega \left(p^\varepsilon(T,x)\right)^2 dx \leqslant \int_\Omega \left(p^0(T,x)\right)^2 dx.$$

It follows that $\|p^\varepsilon(T,\cdot)\|_{L^2(\Omega)}$ converges to $\|p^0(T,\cdot)\|_{L^2(\Omega)}$ as $\varepsilon\searrow 0$, and since $p^\varepsilon(T,\cdot)$ converges to $p^0(T,\cdot)$ weakly in $L^2(\Omega)$, it follows that this convergence is in fact strong, whence the claim.

It remains to investigate the convergence of $J_T^\varepsilon(u^\varepsilon)$ as $\varepsilon\searrow 0$. According to Theorem 2.1 and its proof, one has $n_{in}^\varepsilon(T,\cdot)+n_{un}^\varepsilon(T,\cdot) = K-\varepsilon K n^\varepsilon(T,\cdot)$ and therefore, $(n_{in}^\varepsilon(T,\cdot)+n_{un}^\varepsilon(T,\cdot))_{\varepsilon>0}$ converges to K in $L^\infty(\Omega)$. Since $(p^\varepsilon(T))_{\varepsilon>0}$ converges to $p^0(T,\cdot)$ in $L^2(\Omega)$ as $\varepsilon\searrow 0$, it follows that $(n_{in}^\varepsilon(T,\cdot), n_{un}^\varepsilon(T,\cdot)))_{\varepsilon>0}$ converges to $(Kp^0(T,\cdot), K(1-p^0(T,\cdot)))$ in $L^2(\Omega)$. Since the *Wolbachia*-infected equilibrium $(n_{in,W}^*,0)$ converges to $(K,0)$ as $\varepsilon\searrow 0$, according to (9.4), by passing to the limit in (9.5), it follows that $J_T^\varepsilon(u^\varepsilon)$ converges, as $\varepsilon\searrow 0$ to $J_T^0(u^0)$ given by

$$J_T^0(u^0) = \frac{K^2}{2}(1-p^0(T))^2 + \frac{K^2}{2}(1-p^0(T))^2 = K^2(1-p^0(T))^2,$$

where p^0 denotes the solution of (9.9).

4.2 Analysis of the optimal control problem ($\mathcal{P}_{\text{reduced}}$)

Existence of an optimal control

As a preliminary remark, note that existence of an optimal control has been shown in [3, Theorem 1.1] in a more general setting. To make this article self-contained, we recall the argument hereafter. The analysis to follow is valid under the assumption ($\mathcal{H}_{f,g}$) on f and g. It is not restricted to the particular choice of functions f and g given by (9.10).

Lemma 4.4. *Let $T > 0$, $C > 0$ and $M > 0$. Problem ($\mathcal{P}_{\text{reduced}}$) admits a solution u_0^*.*

Proof. In what follows, we will denote by p_{u_0} the solution to Problem (9.19) associated to the control choice u_0. Let $(u_0^n)_{n\in\mathbb{N}} \in (\mathcal{V}_{C,M})^{\mathbb{N}}$ be a minimizing sequence for Problem ($\mathcal{P}_{\text{reduced}}$). Notice that, since u_0^n belongs to $\mathcal{V}_{C,M}$ and the range of G^{-1} is included in $[0,1[$, we infer from the maximum principle that $0 \leqslant p_{u_0^n}(t,\cdot) < 1$ for a.e. $t \in [0,T]$ so that $(J_T(u_0^n))_{n\in\mathbb{N}}$ is bounded and $\inf_{u_0\in\mathcal{V}_{C,M}} J_T(u_0)$ is finite.

Since the class $\mathcal{V}_{C,M}$ is closed for the L^∞ weak-star topology, there exists $u_0^\infty \in \mathcal{V}_{C,M}$ such that, up to a subsequence, u_0^n converges weakly-star to u_0^∞ in L^∞. Here and in the sequel, we will denote similarly with a slight abuse of notation a given sequence and any subsequence.

Multiplying the main equation of (9.19) by $p_{u_0^n}$ and integrating by parts, we infer from the above estimates the existence of a positive constant C such that

$$\frac{1}{2}\int_0^T\int_\Omega \partial_t(p_{u_0^n}(t,x)^2)dxdt + D\int_0^T\int_\Omega |\nabla p_{u_0^n}(t,x)|^2 dxdt \leqslant C$$

for every $n \in \mathbb{N}$, which also reads

$$\frac{1}{2}\int_\Omega \left[(p_{u_0^n}(t,x))^2)\right]_{t=0}^{t=T} dx + D\int_0^T\int_\Omega |\nabla p_{u_0^n}(t,x)|^2\, dxdt \leqslant C$$

for every $n \in \mathbb{N}$.

By using the pointwise bounds on $p_{u_0^n}$, one gets that $p_{u_0^n}$ is uniformly bounded in $L^2(0,T;H^1(\Omega))$. Furthermore, according to (9.19), the sequence $\partial_t p_{u_0^n}$ is uniformly bounded in $L^2(0,T;W^{-1,1}(\Omega))$. The Aubin-Lions theorem (see [83] and Lemma 4.3) yields that $p_{u_0^n}$ converges (up to a subsequence) to $p^\infty \in L^2(0,T;H^1(\Omega))$, strongly in $L^2(0,T;L^2(\Omega))$ and weakly in $L^2(0,T;H^1(\Omega))$. Furthermore, using that the sequence $\partial_t p_{u_0^n}$ is uniformly bounded in $L^2(0,T;W^{-1,1}(\Omega))$ also yields that $\partial_t p^\infty$ belongs to $L^2(0,T;W^{-1,1}(\Omega))$. Furthermore, reproducing the standard variational argument used in the proof of Corollary 4.1 to show the weak convergence of $p^\varepsilon(T,\cdot)$ to $p^0(T,\cdot)$ in $L^2(\Omega)$ as $\varepsilon \searrow 0$, one shows that for all $t \in [0,T]$, $p_{u_0^n}(t,\cdot)$ also converges weakly, up to a subsequence, to $p^\infty(t,\cdot)$ in $L^2(\Omega)$.

Passing to the limit in (9.19) yields that p^∞ is a weak solution to

$$\begin{cases} \partial_t p^\infty(t,x) - D\Delta p^\infty(t,x) = f(p^\infty(t,x)), & t \in (0,T), \quad x \in \Omega, \\ \partial_\nu p^\infty(t,x) = 0, & t \in (0,T), \quad x \in \partial\Omega. \end{cases}$$

It is standard that any solution to this bistable reaction-diffusion equation is continuous in time.

It remains to show that $u_0^\infty = G(p^\infty(0^+,.))$. Note first that G is convex since g is decreasing on $(0,1)$ under assumption ($\mathcal{H}_{f,g}$). According to the convergence results above, since $p_{u_0^n}(0,\cdot)$ converges weakly (up to a subsequence) to $p^\infty(0,\cdot)$ in $L^2(\Omega)$ and since $u_0^n := G(p_{u_0^n}(0^+,.))$, we get that $G(p^\infty(0,\cdot)) = u_0^\infty$ and hence, $p^\infty = p_{u_0^\infty}$ by passing to the limit as $n \to +\infty$ in the variational formulation on $p_{u_0^n}$.

Finally, let us show that u_0^∞ belongs to $\mathcal{V}_{C,M}$. Since the derivative of G is $1/g$ which is positive, G is increasing and therefore, one has $0 \leqslant u_0^\infty \leqslant M$ a.e. in Ω. Moreover, since $\int_\Omega u_0^n \leqslant C$ rewrites $\langle u_0^n, 1\rangle_{L^\infty,L^1} \leqslant C$, we immediately get that the integral condition is satisfied by u_0^∞.

Therefore, u_0^∞ solves Problem ($\mathcal{P}_{\text{reduced}}$). □

First and second order optimality conditions

We now state first and second order optimality conditions. The objective is twofold: first, we will analyze the optimality of constant solutions, and second, we will use them to derive adapted numerical algorithms.

Definition 4.1. *Let $u_0 \in \mathcal{V}_{C,M}$. A function h in $L^\infty(\Omega)$ is said to be an* ***admissible perturbation*** *of u_0 in $\mathcal{V}_{C,M}$ if, for every sequence of positive real numbers $(\varepsilon_n)_{n\in\mathbb{N}}$ decreasing to 0, there exists a sequence of functions h^n converging to h for the weak-star topology of $L^\infty(\Omega)$ as $n \to +\infty$, and such that $u + \varepsilon_n h^n \in \mathcal{V}_{C,M}$ for every $n \in \mathbb{N}$.*

Proposition 4.5. *Let $u_0 \in \mathcal{V}_{C,M}$ and h be an admissible perturbation. The functional J_T is two times differentiable in the sense of Fréchet at u_0 and one has*

$$dJ_T(u_0)\cdot h = \int_\Omega q(0^+,x)(G^{-1})'(u_0(x))h(x)\,dx,$$

where q denotes the adjoint state, solving the backward p.d.e.

$$\begin{cases} -\partial_t q(t,x) - D\Delta q(t,x) = f'(p(t,x))q(t,x), & (t,x)\in(0,T)\times\Omega,\\ \partial_\nu q(t,x) = 0, & (t,x)\in(0,T)\times\partial\Omega,\\ q(T,x) = p(T,x) - 1 & x\in\Omega \end{cases} \tag{9.26}$$

and p denotes the solution to (9.19) associated to the control choice u_0.

Furthermore, the second order derivative of J_T at u_0 reads

$$\begin{aligned} d^2J_T(u_0)(h,h) &= -\int_\Omega\int_0^T \dot p(t,x)^2 f''(p(t,x))q(t,x)\,dtdx + \int_\Omega \dot p(T,x)^2\,dx \\ &\quad + \int_\Omega q(0^+,x)(G^{-1})''(u_0(x))h(x)^2\,dx \end{aligned}$$

for every admissible perturbation h, where $\dot p$ denotes the solution to the linear system

$$\begin{cases} \partial_t \dot p(t,x) - D\Delta \dot p(t,x) = \dot p(t,x) f'(p(t,x)) & (t,x)\in(0,T)\times\Omega,\\ \partial_\nu \dot p(t,x) = 0 & (t,x)\in(0,T)\times\partial\Omega,\\ \dot p(0^+,x) = (G^{-1})'(u_0)h(x) & x\in\Omega. \end{cases} \tag{9.27}$$

Proof. As a preliminary remark, we claim that for any element u of the set $\mathcal{V}_{C,M}$ and any admissible perturbation h, the mapping $u \in \mathcal{V}_{C,M} \mapsto p \in L^2(0,T;H^1(\Omega))$, where p_u denotes the unique weak solution of (9.19), is differentiable in the sense of Gâteaux at u in direction h. Indeed, proving such a property is standard in calculus of variations and rests upon the implicit function theorem.

Let $u \in \mathcal{V}_{C,M}$. Let h denote an admissible perturbation. Observing that $p_{u+\varepsilon h}$ solves the system

$$\begin{cases} \partial_t p_{u+\varepsilon h}(t,x) - D\Delta p_{u+\varepsilon h}(t,x) = f(p_{u+\varepsilon h}(t,x)), & (t,x) \in (0,T)\times\Omega, \\ \partial_\nu p_{u+\varepsilon h}(t,x) = 0, & (t,x) \in (0,T)\times\partial\Omega, \\ p_{u+\varepsilon h}(0^+,x) = G^{-1}(u+\varepsilon h(x)), & x \in \Omega. \end{cases}$$

Let $\dot{p}$ denote the derivative of $\varepsilon \mapsto p(u+\varepsilon h)$ at $\varepsilon = 0$. Standard computations yield that $\dot{p}$ solves the linearized reaction-diffusion system

$$\begin{cases} \partial_t \dot{p}(t,x) - D\Delta \dot{p}(t,x) = \dot{p}(t,x) f'(p_u(t,x)), & (t,x) \in (0,T)\times\Omega, \\ \partial_\nu \dot{p}(t,x) = 0, & (t,x) \in (0,T)\times\partial\Omega, \\ \dot{p}(0^+,x) = (G^{-1})'(u(x))h(x), & x \in \Omega. \end{cases} \tag{9.28}$$

Furthermore, according to the chain rule, one has

$$dJ_T(u)\cdot h = \lim_{\varepsilon\to 0} \frac{J_T(u+\varepsilon h) - J_T(u)}{\varepsilon} = \int_\Omega \dot{p}(T,x)\left(p_u(T,x) - 1\right)\, dx.$$

Let us multiply the main equation of (9.28) by q_u, and integrate then two times by parts on $(0,T)\times\Omega$. One thus gets

$$\int_0^T \int_\Omega \partial_t \dot{p}(t,x) q_u(t,x)\, dxdt = \int_0^T \int_\Omega D\dot{p}(t,x)\Delta q_u(t,x)\, dxdt + \int_0^T \int_\Omega \dot{p}(t,x) f'(p_u(t,x)) q_u(t,x)\, dxdt. \tag{9.29}$$

Similarly, let us multiply the main equation of (9.26) by $\dot{p}$, and integrate then by parts on $(0,T)\times\Omega$. We obtain

$$-\int_0^T \int_\Omega \dot{p}(t,x)\partial_t q_u(t,x)\, dxdt = \int_0^T \int_\Omega D\dot{p}(t,x)\Delta q_u(t,x)\, dxdt + \int_0^T \int_\Omega \dot{p}(t,x) f'(p_u(t,x)) q_u(t,x)\, dxdt. \tag{9.30}$$

By comparing (9.29) and (9.30), we infer that

$$\int_0^T \int_\Omega \left(\dot{p}(t,x)\partial_t q_u(t,x) + \partial_t \dot{p}(t,x) q_u(t,x)\right)\, dxdt = 0$$

leading to the following duality identity:

$$\int_\Omega \left(\dot{p}(T,x) q_u(T,x) - \dot{p}(0,x) q_u(0,x)\right) dx = 0.$$

By using (9.28) and (9.26), we rewrite the expression above as

$$\int_\Omega \dot{p}(T,x)(p_u(T,x)-1) = \int_\Omega (G^{-1})'(u(x))h(x)q_u(0,x)dx.$$

Thus the desired expression of the derivative follows.

Let us now compute $d^2J_T(u_0)$. Since J_T is two times differentiable, one has

$$\begin{aligned}
&d^2J_T(u_0)(h,h)\\
&= \lim_{\varepsilon\to 0} \frac{dJ_T(u_0+\varepsilon h)\cdot h - dJ_T(u_0)\cdot h}{\varepsilon}\\
&= \lim_{\varepsilon\to 0} \frac{\int_\Omega q_{u_0+\varepsilon h}(0^+,x)(G^{-1})'(u_0+\varepsilon k)(x)h(x)\,dx - \int_\Omega q_{u_0}(0^+,x)(G^{-1})'(u_0(x))h(x)\,dx}{\varepsilon}\\
&= \int_\Omega \dot{q}(0^+,x)(G^{-1})'(u_0(x))h(x)\,dx + \int_\Omega q_{u_0}(0^+,x)h(x)^2(G^{-1})''(u_0(x))\,dx,
\end{aligned}$$

where $\dot{q}$ is given by

$$\dot{q}(t,x) = \lim_{\varepsilon\to 0} \frac{q_{u_0+\varepsilon h}(t,x) - q_{u_0}(t,x)}{\varepsilon}.$$

A standard reasoning enables us to prove that $\dot{q}$ solves the linear p.d.e.

$$\begin{cases}
-\partial_t\dot{q}(t,x) - D\Delta\dot{q}(t,x) & \\
\qquad = \dot{p}(t,x)f''(p_u(t,x))q(t,x) + f'(p_u(t,x))\dot{q}(t,x), & (t,x)\in(0,T)\times\Omega,\\
\partial_\nu\dot{q}(t,x) = 0, & (t,x)\in(0,T)\times\partial\Omega,\\
\dot{q}(T,x) = \dot{p}(T,x), & x\in\Omega,
\end{cases} \tag{9.31}$$

with $\dot{p}$, the solution of the linear p.d.e. (9.27). One has

$$\begin{aligned}
&\int_\Omega \dot{q}(0^+,x)(G^{-1})'(u_0(x))h(x)\,dx\\
&= \int_\Omega \dot{q}(0^+,x)\dot{p}(0^+,x)\ dx\\
&= \int_\Omega \dot{q}(0^+,x)\dot{p}(0^+,x)\ dx - \int_\Omega \dot{q}(T,x)\dot{p}(T,x)\ dx + \int_\Omega \dot{q}(T,x)\dot{p}(T,x)\ dx\\
&= \int_\Omega\int_0^T \partial_t\dot{p}(t,x)\dot{q}(t,x)dtdx + \int_\Omega\int_0^T \partial_t\dot{q}(t,x)\dot{p}(t,x)dtdx + \int_\Omega \dot{p}(T,x)^2\ dx.
\end{aligned}$$

By using the main equation in Systems (9.31) and (9.27), one gets

$$\begin{aligned}
\int_\Omega \dot{q}(0^+,x)(G^{-1})'(u_0(x))h(x)\,dx = &\int_\Omega\int_0^T \left[D\Delta\dot{p}(t,x) + \dot{p}(t,x)f'(p_u(t,x))\right]\dot{q}(t,x)\,dtdx\\
&+ \int_\Omega\int_0^T \left[-D\Delta\dot{q}(t,x) - \dot{p}(t,x)f''(p_u(t,x))q_u(t,x)\right.\\
&\left.-f'(p_u(t,x))\dot{q}(t,x)\right]\dot{p}(t,x)\,dtdx + \int_\Omega \dot{p}(T,x)^2\ dx.
\end{aligned}$$

The Green formula finally yields

$$\int_\Omega \dot{q}(0^+,x)(G^{-1})'(u_0(x))h(x)\,dx = \int_\Omega\int_0^T -\dot{p}(t,x)^2f''(p_u(t,x))q_u(t,x)dtdx + \int_\Omega \dot{p}(T,x)^2\ dx,$$

whence the expected expression for the second order derivative. □

Let us now derive first and second order optimality conditions for this problem.

Proposition 4.6 (Necessary first and second orders optimality conditions)**.** *For all $u_0 \in \mathcal{V}_{C,M}$ consider $\psi[u_0]$ denote the function defined on Ω by*

$$\psi[u_0](\cdot) = q(0^+, \cdot)(G^{-1})'(u_0(\cdot)),$$

where q solves the adjoint system (9.26) associated to the control choice u_0.

Let u_0^* be a solution to Problem ($\mathcal{P}_{\text{reduced}}$). Then, there exists $\lambda \in [0, +\infty)$ such that

$$\begin{array}{ll} \text{on } \{u_0^* = M\}, & \psi[u_0^*] \leqslant -\lambda, \\ \text{on } \{u_0^* = 0\}, & \psi[u_0^*] \geqslant -\lambda, \\ \text{on } \{0 < u_0 < M\}, & \psi[u_0^*] = -\lambda, \end{array} \tag{9.32}$$

(called necessary first order optimality condition) or equivalently, the function Λ defined by

$$\Lambda : x \in \Omega \mapsto \min\{u_0^*(x), \max\{u_0^*(x) - M, \psi[u_0^*](x) + \lambda\}\}$$

vanishes identically in Ω. Moreover, one has $\lambda \left(\int_\Omega u_0^*(x)\, dx - C\right) = 0$ (slackness condition).

Moreover, the second order optimality conditions for this problem read: $d^2 J_T(u_0^*)(h, h) \geqslant 0$ for every admissible perturbation h such that $dJ_T(u_0^*) \cdot h = 0$.

Proof. Let us introduce the Lagrangian functional associated to Problem ($\mathcal{P}_{\text{reduced}}$), given by

$$\mathcal{L} : (u, \lambda) \in \mathcal{V}_{C,M} \times \mathbb{R}_+ \mapsto J_T(u) + \lambda \left(\int_\Omega u - C \right).$$

According to Proposition 4.5, and denoting by d_{un} the differential operator with respect to the variable u, the Euler inequation associated to Problem ($\mathcal{P}_{\text{reduced}}$) reads: $d_{un}\mathcal{L}(u, \lambda) \cdot h \geqslant 0$ for all admissible perturbation h of u_0^* in $\{u_0 \in L^\infty(0, T),\ 0 \leqslant u_0 \leqslant M \text{ a.e. in } \Omega\}$. This can be rewritten

$$\int_\Omega (\psi[u_0^*](x) + \lambda)\, h(x)\, dx \geqslant 0$$

for all functions h as above. The analysis of such optimality condition is standard in optimal control theory (see for example [66]) and yields:

$$\begin{array}{ll} \text{on } \{u_0^* = M\}, & \psi[u_0^*] \leqslant -\lambda, \\ \text{on } \{u_0^* = 0\}, & \psi[u_0^*] \geqslant -\lambda, \\ \text{on } \{0 < u_0 < M\}, & \psi[u_0^*] = -\lambda. \end{array}$$

Moreover, one has $\lambda \left(\int_\Omega u_0^*(x)\, dx - C\right) = 0$ (slackness condition). It remains to show that such conditions also rewrite $\Lambda(\cdot) = 0$ in Ω. It is straightforward that if the optimality conditions above are satisfied, then $\Lambda(\cdot) = 0$ in Ω. Let us examine the converse sense, assuming that $\Lambda(\cdot) = 0$ in Ω. Then, for a.e. $x \in \{u_0^* = 0\}$, one has

$$\max\{u_0^*(x) - M, \psi[u_0^*](x) + \lambda\} = \max\{-M, \psi[u_0^*](x) + \lambda\} \geqslant 0$$

and thus, $\psi[u_0^*](x) \geqslant -\lambda$. The analysis is exactly similar on the set $\{u_0^* = M\}$. Finally, if x denotes a Lebesgue point of the $\{0 < u_0^* < M\}$, one has necessarily

$$\max\{u_0^*(x) - M, \psi[u_0^*](x) + \lambda\} = 0$$

and therefore, $\psi[u_0^*](x) = -\lambda$. This concludes the first part of this proposition. The second part is standard (see e.g. [57]). □

We infer from this result that either the pointwise or the integral constraint is saturated by every minimizer u_0^*.

Corollary 4.7. *Let u_0^* be a solution to Problem ($\mathcal{P}_{\text{reduced}}$). Then, one has necessarily*

$$\int_\Omega u_0^*(x)\, dx = \min\{C, M|\Omega|\}.$$

Proof. Let us first assume that $M \geqslant C/|\Omega|$. Let us argue by contradiction, assuming that $\int_\Omega u_0^* < C$. Let p (resp. q) denote the solution to the direct problem (9.19) (resp. the adjoint problem (9.26)) associated to the control choice u_0^*. According to Theorem 4.6 and its proof, the slackness condition implies that $\lambda = 0$. Recall that one has $p(t,x) \in (0,1)$ for a.e. $(t,x) \in (0,T) \times \Omega$, as highlighted in Section 2.3, and therefore $q(T,\cdot) \in (-1,0)$ a.e. in Ω. A simple comparison argument yields that q is negative in $(0,T) \times \Omega$ (see e.g. [21]). Since G is bijective and increasing, so is G^{-1} and we infer that ψ is negative in Ω. By using Theorem 4.6, we get that necessarily, $u_0^*(\cdot) = M$, which is in contradiction with the assumption above on M and C.

The case where $M < C/|\Omega|$ is solved hereafter, in the proof of Theorem 3.1. □

Optimality of constant solutions

This section is devoted to the proof the our main results, that is Therem 3.1. Let us first show (i). The proof rests upon a simple comparison argument: one shows more precisely that $\overline{u}_M$ solves Problem ($\mathcal{P}_{\text{reduced}}$) as soon as it belongs to $\mathcal{V}_{C,M}$ which is equivalent to the condition above on the parameters.

Let $u \in \mathcal{V}_{C,M}$. Let p and p_M denote the solutions to System (9.19) corresponding respectively to the control choices u and $\overline{u}_M$.

Since u belongs to $\mathcal{V}_{C,M}$ and G^{-1} is increasing, one has $G^{-1}(u(x)) \leqslant G^{-1}(M)$ for a.e. $x \in \Omega$, meaning that $p(0^+,\cdot) \leqslant p_M(0^+,\cdot)$ on Ω. According to the parabolic comparison principle, we infer that $p(t,\cdot) \leqslant p_M(t,\cdot)$ on Ω, for all $t \in [0,T)$, so that one gets in particular that $p^*(T,\cdot) \leqslant p_M(T,\cdot)$ in Ω, and therefore, $J_T(\overline{u}_M) \leqslant J_T(u)$. Uniqueness follows from the monotonicity of G and the comparison principle, since $0 \leqslant u \leqslant M$ a.e. in Ω.

Let us now prove (ii). Set $c = C/|\Omega|$. According to the optimality conditions (9.32), since $c < M$, the function $\overline{u}$ identically equal to the constant c satisfies the first order optimality conditions if, and only if, there exists $\lambda \in \mathbb{R}_+$ such that $\psi(\cdot) = -\lambda$ in Ω. Since $(G^{-1})'(\overline{u}(\cdot))$ is constant in Ω, this is equivalent to say that $q(0^+,\cdot)$ is constant in Ω.

First, observe that, by uniqueness of the solutions to the reaction-diffusion system (9.19), the associated solution $\overline{p}$ is constant in space. Moreover, writing $\overline{p}(t,\cdot) = \overline{p}(t)$ with a slight abuse of notation, one easily sees that $\overline{p}$ solves the ODE

$$\begin{cases} \overline{p}'(t) = f(\overline{p}(t)), & t \in [0,T], \\ \overline{p}(0^+) = G^{-1}(c). \end{cases} \tag{9.33}$$

Standard uniqueness arguments coming from the Cauchy-Lipschitz theorem show that if $\overline{p}(0^+) \notin \{0,\theta,1\}$ (the set of roots of f), then $f(\overline{p}(\cdot))$ does not vanish on $[0,T]$ and has hence a constant sign.

Note that, since $c \neq 0$, one cannot have $\overline{p}(0^+) = 0$. Similarly, noting that G is an increasing bijection from $[0,1)$ into $[0,+\infty)$, we infer that one cannot have $\overline{p}(0^+) = 1$. Let $\overline{p}(0^+) \in (0,\theta) \cup (\theta,1)$. Then, $f(\overline{p}(0^+)) \neq 0$, and using that $\overline{p}$ has a constant sign, which allows us to write

$$\overline{p}'(t) = f(\overline{p}(t)) \Rightarrow \forall t \in [0,T], \quad \int_{\overline{p}(0)}^{\overline{p}(t)} \frac{1}{f(u)} du = t$$

and therefore,

$$\overline{p}(t) = F^{-1}(t + F(G^{-1}(\overline{u}))),$$

for all $t \in [0,T]$, where F denotes an antiderivative of $1/f$. Indeed, since f has a constant sign, F is monotone and continuous, whence the existence of F^{-1}.

Proceeding similarly for the solution $\overline{q}$ to System (9.26) associated to $p = \overline{p}$ drives us to look for constant solutions with respect to the space variable. Let $\overline{q}$ denote such a solution (whenever it exists). Hence, it solves

$$\begin{cases} \overline{q}'(t) = -f'(\overline{p}(t))\overline{q}(t), & t \in [0,T], \\ \overline{q}(T) = \overline{p}(T) - 1 \end{cases}$$

and therefore,

$$\overline{q}(t) = (\overline{p}(T) - 1) \exp\left(\int_t^T f'(\overline{p}(s)) ds \right).$$

By uniqueness of the solution to (9.26), it follows that $\overline{q}$ solves (9.26).

Now, if $\overline{p}(0^+) = \theta$, meaning that $\overline{u} = G(\theta)$, then $\overline{p}(\cdot) = \theta$ and one has $\overline{q}(t) = (\theta - 1)e^{(T-t)f'(\theta)}$ for all $t \in [0,T]$.

All in all, we get that $\overline{q}(0^+,\cdot)$ is constant on Ω and the switching function ψ, which is constant, reads

$$\psi(\cdot) = (G^{-1})'(c)(\overline{p}(T) - 1) \exp\left(\int_0^T f'(\overline{p}(s)) ds \right) \leqslant 0,$$

by using that $\overline{p}(t) \in (0,1)$ for all $t \in [0,T]$ and that G is bijective and increasing. We infer that the first order optimality conditions are satisfied by $\overline{u}$.

To investigate the second order optimality conditions, it is convenient to introduce the Hilbert basis $\{w_n\}_{n\in\mathbb{N}^*}$ of $L^2(\Omega)$ made of the Neumann-Laplacian eigenfunctions defined by:

$$w_1(\cdot) = \frac{1}{|\Omega|}, \text{and for } n \geqslant 2,\ w_n \text{ solves the p.d.e. } \begin{cases} -D\Delta w_n = \lambda_n w_n, & \text{in } \Omega, \\ \partial_n w_n = 0, & \text{on } (0,T)\times\partial\Omega, \\ \int_\Omega w_n(x)\,dx = 0, \\ \|w_n\|_{L^2(\Omega)} = 1, \end{cases}$$

where $(\lambda_n)_{n\geqslant 2}$ denotes the sequence of associated positive eigenvalues.

In this setting, let us expand every admissible perturbation h as

$$h = \sum_{n=1}^{+\infty} \alpha_n w_n \quad \text{with } \alpha_n = \langle h, w_n\rangle_{L^2(\Omega)} \text{ for all } n \in \mathbb{N}^*.$$

Using that the solution $\overline{p}$ to (9.19) does not depend on the space variable, it is standard to expand $\dot{p}$ as

$$\dot{p}(t,x) = \sum_{n=1}^{\infty} \alpha_n v_n(t) w_n(x) \text{ for each } t \in (0,T), \quad x \in \Omega,$$

where v_n solves the o.d.e. $v_n'(t) = (-\lambda_n + f'(\overline{p}(t)))\, v_n(t)$ and $v_n(0) = (G^{-1})'(c)$ so that

$$v_n(t) = (G^{-1})'(c) \exp\left(-\lambda_n t + \int_0^t f'(\overline{p}(s))ds\right).$$

According to Proposition 4.5, one thus computes

$$\begin{aligned} d^2 J_T(\overline{u})(h,h) &= \int_\Omega \dot{p}(T,x)^2\,dx - \int_\Omega\int_0^T \dot{p}(t,x)^2 f''(\overline{p}(t))\overline{q}(t)\,dtdx \\ &+ \int_\Omega \overline{q}(0)(G^{-1})''(c)h(x)^2\,dx \\ &= \int_\Omega \left(\sum_{n=1}^{+\infty} \alpha_n v_n(T) w_n(x)\right)^2 dx - \int_\Omega\int_0^T \left(\sum_{n=1}^{+\infty} \alpha_n v_n(t) w_n(x)\right)^2 f''(\overline{p}(t))\overline{q}(t)\,dtdx \\ &+ \int_\Omega \overline{q}(0)(G^{-1})''(c)\left(\sum_{n=1}^{+\infty} \alpha_n w_n(x)\right)^2 dx. \end{aligned}$$

Using that $\{w_n\}_{n\in\mathbb{N}^*}$ is orthonormal in $L^2(\Omega)$, we finally get the following diagonalized expression of the second order derivative

$$d^2 J_T(\overline{u})(h,h) = \sum_{n=1}^{+\infty} \delta_n(T)\alpha_n^2,$$

$$\text{with } \delta_n(T) = v_n(T)^2 - \int_0^T f''(\overline{p}(t))\overline{q}(t)v_n(t)^2 dt + \overline{q}(0)(G^{-1})''(c).$$

The signature of $d^2 J_T(\overline{u})(h,h)$ seen as an infinite quadratic form with respect to h is then directly given by the sign of the coefficients δ_n. Notice that for all $n \in \mathbb{N}^*$, one has

$$\delta_n(T) \;=\; v_n(T)^2 + (\overline{p}(T)-1)e^{\int_0^T f'(\overline{p})}\left[(G^{-1})''(c) - \int_0^T f''(\overline{p}(t))e^{-\int_0^t f'(\overline{p})} v_n(t)^2\,dt\right].$$

Let us first assume that $C \leqslant |\Omega| G(\theta)$, meaning that $(G^{-1})(c) \leqslant \theta$. In that case, since $\overline{p}$ solves (9.33), and that the three roots of f are 0, θ and 1, one infers that $\overline{p}$ is a decreasing function and that $f(\overline{p})$ remains negative all along $(0,T)$. Furthermore, on $(0,\theta)$, the function f'' is positive. Finally, one computes $(G^{-1})''(c) = (G^{-1})'(c) g'(G^{-1}(c))$ which is negative since so is g' on $(0,1)$. Combining all these facts, we infer that

$$(G^{-1})''(c) - \int_0^T f''(\overline{p}(t)) e^{-\int_0^t f'(\overline{p})} v_n(t)^2 \, dt < 0$$

and since $\overline{p}(T) < 1$, it follows that

$$\delta_n(T) > (\overline{p}(T) - 1)(G^{-1})''(c) e^{\int_0^T f'(\overline{p})} > 0$$

for every $n \in \mathbb{N}^*$. Therefore, by setting

$$K_T = (\overline{p}(T) - 1)(G^{-1})''(c) e^{\int_0^T f'(\overline{p})} > 0,$$

we get that for every admissible perturbation h, one has

$$d^2 J_T(\overline{u})(h,h) \geqslant K_T \sum_{n=1}^{+\infty} \alpha_n^2 = K_T \|h\|^2_{L^2(\Omega)}.$$

Expanding J_T at the second order at $\overline{u}$, it is then standard that this condition implies that $\overline{u}$ is a local minimizer for the functional J_T.

Constant solutions are not always global minimizers

We recall the following well-known result (see [76]).

Lemma 4.8. *Let $\alpha \in (\theta_c, 1)$. There exists a unique solution, denoted v_α, of the Cauchy problem*

$$-\frac{d-1}{r} v'(r) - v''(r) = f(v(r)), \text{ on } (0,+\infty), \quad v(0) = \alpha, \quad v'(0) = 0.$$

Moreover, $r \in (0,+\infty) \mapsto v_\alpha(r)$ is decreasing, and there exists $R_\alpha > 0$ such that $v_\alpha(R_\alpha) = 0$.

In other words, this lemma states the existence of radially symmetric steady-states to the stationary equation associated to (9.19). We then deduce the existence of stationary subsolutions for System (9.19) that are positive and compactly supported, provided the domain contains a large enough ball, in other words that the inradius of Ω be large enough.

Corollary 4.9. *Let us assume that a ball of radius R_α is compactly included in Ω for some $\alpha \in (\theta_c, 1)$, in other words that there exists $O_\alpha \in \Omega$ such that $\overline{B(O_\alpha, R_\alpha)} \subset \Omega$. Then, $w_\alpha := \max\{0, v_\alpha(\|x - O_\alpha\|)\}$ is a subsolution of (9.19) if, and only if $G(w_\alpha) \leqslant u_0$.*

Using that w_α is a subsolution, we deduce the following comparison result.

Corollary 4.10. *For any $\alpha \in (\theta_c, 1)$ such that Ω contains strictly a ball of radius R_α, that is there exists $O_\alpha \in \Omega$ such that $\overline{B(O_\alpha, R_\alpha)} \subset \Omega$, and $G(w_\alpha) \leqslant u_0$, the solution of (9.19) verifies $p(t,\cdot) \geqslant w_\alpha$ on Ω for any $t \geqslant 0$.*

Let us introduce

$$C_\alpha := \int_\Omega G(w_\alpha(x))\,dx.$$

Notice that the family of subsolutions $(w_\alpha)_\alpha$ have already been used to provide a sufficient condition on the release function to initiate propagation of infected mosquitoes [90].

Remark 4.1. *It is worth mentioning that in the one dimensional case, the expressions for R_α and C_α are completely explicit:*

$$R_\alpha = \int_0^\alpha \frac{dw}{\sqrt{2(F(\alpha)-F(w))}}, \qquad C_\alpha = \int_0^\alpha \frac{2G(w)\,dw}{\sqrt{2(F(\alpha)-F(w))}}.$$

We are now in position to prove Proposition 3.2 that we rewrite more precisely using the notations above.

Proposition 4.11. *Let us assume (9.20). Assume moreover the existence of $\alpha \in (\theta_c, G^{-1}(M)]$ such that Ω contains strictly a ball of radius R_α, and $C_\alpha \leqslant C$. Then the constant solution $\overline{u} := \frac{C}{|\Omega|}$ is not a global minimizer of the optimization problem $\mathcal{P}_{reduced}$ whenever T is large enough.*

Proof. From assumption (9.20), we have $G^{-1}(\overline{u}) < \theta$, hence we have already seen in Section 4.2 that the solution, denoted $\overline{p}$, of (9.19) with initial data $G^{-1}(\overline{u})$ is constant in space and decreasing with respect to time. More precisely, it solves the ODE

$$\overline{p}' = f(\overline{p}), \qquad \overline{p}(0) = G^{-1}(\overline{u}) = G^{-1}\Big(\frac{C}{|\Omega|}\Big).$$

Hence, when $t \to +\infty$, $\overline{p}(t)$ decays to 0.

For any $\alpha \in (\theta_c, G^{-1}(M)]$ satisfying the assumptions above, the subsolution w_α defined in Corollary 4.9 is such that $G(w_\alpha) \in \mathcal{V}_{C,M}$. From Corollary 4.10, if we take $u_0 \in \mathcal{V}_{C,M}$ such that $G^{-1}(u_0) \geqslant w_\alpha$, then for all $t \geqslant 0$, the corresponding solution to (9.19) verifies $p(t,\cdot) \geqslant w_\alpha$. Hence $J_T(u_0) \leqslant \frac{1}{2}\int_\Omega (1-w_\alpha(x))^2\,dx$.

Moreover, since $\overline{p}(T) \to 0$ as $T \to +\infty$, we have that for T large enough

$$\int_\Omega (1-\overline{p}(T))^2\,dx = (1-\overline{p}(T))^2|\Omega| > \int_\Omega (1-w_\alpha(x))^2\,dx.$$

Hence, $\overline{u}$ is not a global minimum of J_T at time T since $J_T(u_0) < J_T(\overline{u}) = \frac{1}{2}\int_\Omega (1-\overline{p}(T))^2\,dx$. □

Remark 4.2. *If $G^{-1}(M) > \theta_c$, if the inradius of Ω is large enough and if C is large enough, it is always possible to find α satisfying the assumptions of Proposition 4.11. For instance, it suffices to choose $\alpha = G^{-1}(M)$ and to take $C \geqslant C_{G^{-1}(M)}$ and the inradius of Ω large enough so that (9.20) holds and a ball of radius $R_{G^{-1}(M)}$ be included in Ω.*

5 Numerical experiments

In this section, we provide some numerical approximations of solutions for the optimal control problem ($\mathcal{P}_{\text{reduced}}$).

The parameter values are given in Tables 9.1 and 9.2. We will assume that Ω is an interval $(0, L)$, i.e. $d = 1$. From these tables, we deduce that $s_f = 0.1$, $\delta = \frac{4}{3}$, and thus $\theta = \frac{s_f+\delta-1}{\delta s_h} = \frac{13}{36}$. System (9.19) will be discretized with an explicit Euler scheme in time and a standard finite difference approximation of the Laplacian. In all simulations, the number of steps in space and time will be fixed to 20 and 200 respectively (in order to satisfy the CFL condition). The solution of the optimal control problem will be obtained by testing and combining two approaches:

- a Uzawa type algorithm, based on the gradient computation of Prop. 4.5. It consists in alternating at each iteration a step of minimization of the Lagrangian associated with the problem with respect to the primal variable (u_0) and a step of maximization with respect to the Lagrange multiplier associated with the integral constraint. The minimization step is performed with a projected gradient type method, where L^∞ constraints on u_0 are taken into account by means of a projection operator.
- the opensource optimization routine GEKKO (see [9]) solving the optimization problem using the IPOPT (Interior Point OPTimizer) library, a software package for large-scale nonlinear problems by an interior-point filter line-search algorithm (see [93]). This algorithm has been initialized with the previous control obtained by using the aforementioned Uzawa type algorithm.

Parameter	*Name*	*Value*
F_{un}	Normalized fecundity rate for uninfected mosquitoes	1
F_{in}	Normalized fecundity rate for infected mosquitoes	0.9
d_{un}	Death rate for uninfected mosquitoes	0.27
d_{in}	Death rate for infected mosquitoes	0.36
K	Caring capacity	0.06
s_h	Cytoplasmic incompatibility	0.9

Table 9.1: Values of the parameters used in the simulations (see [90, sec. 2])

Parameter	*Name*	*Value*
T	Time of experiment	40
D	Diffusion coefficient	1
$L = \lvert\Omega\rvert$	Size of the spatial domain	30

Table 9.2: Values of the parameters T, D and $|\Omega|$ used in the simulations

Remark 5.1. *According to Remark 2.1, the assumption ($\mathcal{H}_{f,g}$) is satisfied for the particular choices of functions f and g given by (9.10) under (9.11) and (9.12) which hold true for the values of the parameters in Table 9.1. Furthermore, it is easy to check numerically that the assumption (9.13) is satisfied for the values of the parameters taken from the case at hand*

(see e.g. [90]). Indeed, for the parameter values, $f < 0$ on $(0, \theta)$, which implies that $F < 0$ on $(0, \theta)$, and moreover $F(1) > 0$ (see Figure 9.1).

Let us distinguish between two cases:

$$\textit{Case } C/|\Omega| > M.$$

In Figure 9.2, the local minimizers of Problem ($\mathcal{P}_{\text{reduced}}$) for $C = 1.2$ and $M = 0.02$ (left) (resp. $M = 0.03$ (right)) obtained by using the aformentioned Uzawa and Gekko algorithms are reported. We observe the extinction (resp. the invasion) of the population. One recovers the theoretical result stated in item (i) of Theorem 3.1, in other words that the constant function equal to M solves Problem ($\mathcal{P}_{\text{reduced}}$) whenever $C \geqslant M|\Omega|$ (see Table 9.3). In this situation, the space dependency has no impact on the time dynamics, i.e. the dynamics is the same as if there is no diffusion. Then, since it is a bistable dynamics, when $M < G(\theta)$ there is extinction of the population, whereas there is invasion when $M > G(\theta)$.

$$\textit{Case } C/|\Omega| < M.$$

This situation is illustrated in Figure 9.3 and 9.4 with Gekko algorithm and Figure 9.5 with Uzawa algorithm for $C \in \{0.5, 0.8\}$ and $M \in \{0.04, 0.4\}$. We can see in Figure 9.3 that when the number total of mosquitoes released is too low (when $C = 0.5$), then the infected population decreased until the extinction of this population. On the contrary, if the number total of mosquitoes released is higher (when $C = 0.8$), then we obtain an invasion of the infected mosquitoes. The simulation with the Uzawa algorithm in Figure 9.5 recovers the fact that C/L is a local minimizer for Problem ($\mathcal{P}_{\text{reduced}}$). Indeed this algorithm seems to converges always to this constant solution. Nevertheless, it is not a global minimum since Gekko provides a better control as it is illustrated thanks to the values of $J_T(\overline{u})$ reported in Table 9.3. Moreover, we see on Figure 9.3 that invasion of the infected population seems to occurs whereas the infected population seems to go to extinction in Figure 9.5. This is also in concordance with the result stated in Proposition 4.11.

Case	*Parameters*	$J_T(\overline{u})$ with Gekko	$J_T(\overline{u})$ with Uzawa	$J_T(M)$	$J_T(C/L)$
$C/\lvert\Omega\rvert > M$	$M = 0.02$, $C = 1.2$	14.7	14.7	14.7	
	$M = 0.03$, $C = 1.2$	3.61e-2	3.61e-2	3.61e-2	
$C/\lvert\Omega\rvert < M$	$M = 0.04$, $C = 0.5$	14.0	14.8		14.8
	$M = 0.04$, $C = 0.8$	2.30	12.7		12.7
	$M = 0.08$, $C = 0.5$	13.8	14.8		14.8
	$M = 0.08$, $C = 0.8$	2.25	12.7		12.7

Table 9.3: Values of local optima computed thanks to Gekko and Uzawa algorithms and theoretical local optima

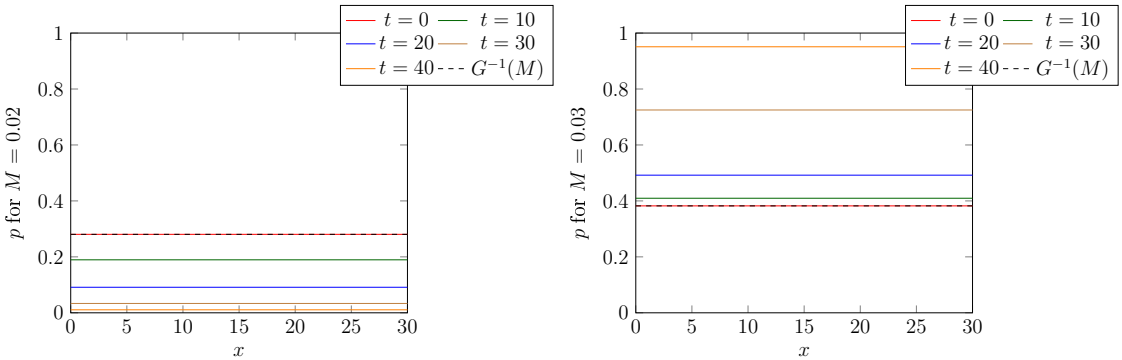

Figure 9.2: Case $C/|\Omega| > M$: Optimal solution p to Problem ($\mathcal{P}_{\text{reduced}}$) at time $t \in \{0, 10, 20, 30, 40\}$ for $C = 1.2$ and $M \in \{0.02, 0.03\}$ thanks to Gekko and Uzawa algorithms

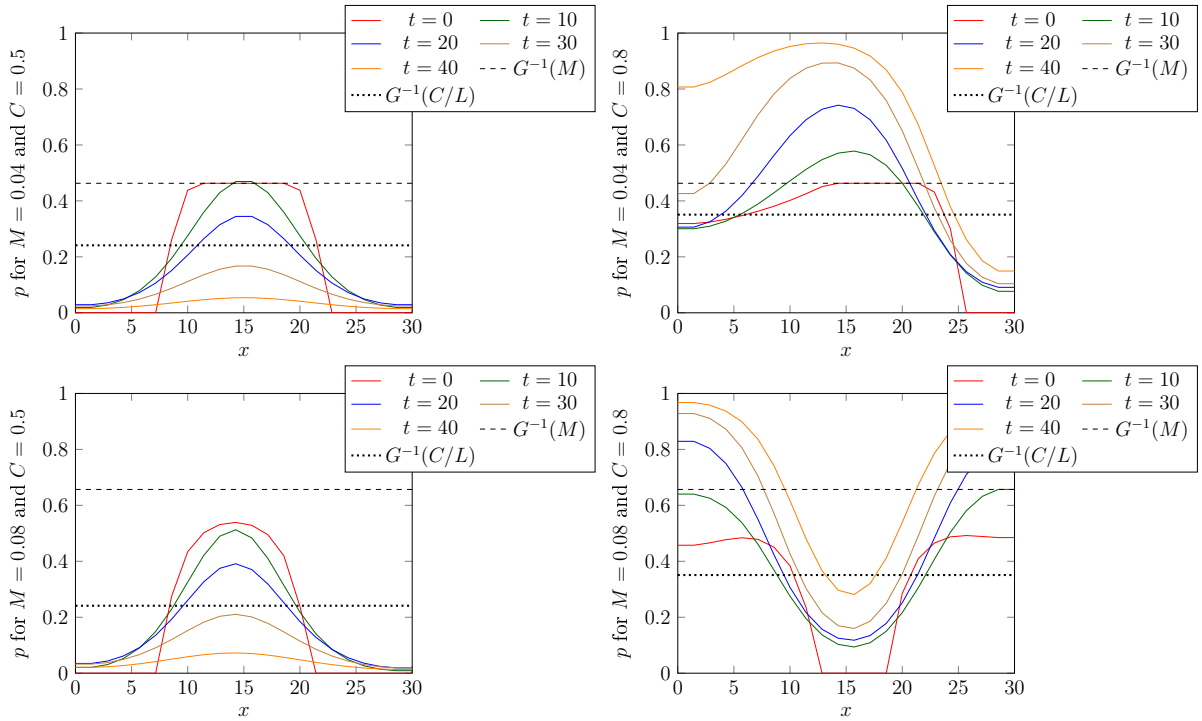

Figure 9.3: Case $C/|\Omega| < M$: Optimal solution p to Problem ($\mathcal{P}_{\text{reduced}}$) at time $t \in \{0, 10, 20, 30, 40\}$ for $C \in \{0.5, 0.8\}$ and $M \in \{0.04, 0.08\}$ thanks to Gekko algorithm.

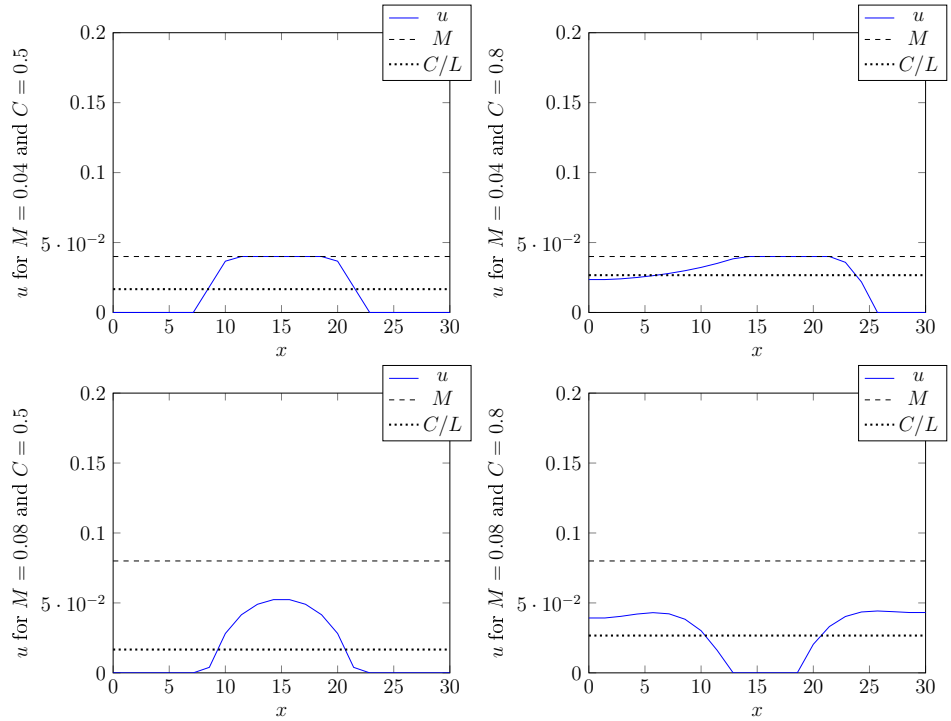

Figure 9.4: Case $C/|\Omega| < M$: Optimal control u associated to the cases considered in Fig. 9.3, in other words solution of Problem ($\mathcal{P}_{\text{reduced}}$) for $C \in \{0.5, 0.8\}$ and $M \in \{0.04, 0.08\}$ thanks to Gekko algorithm

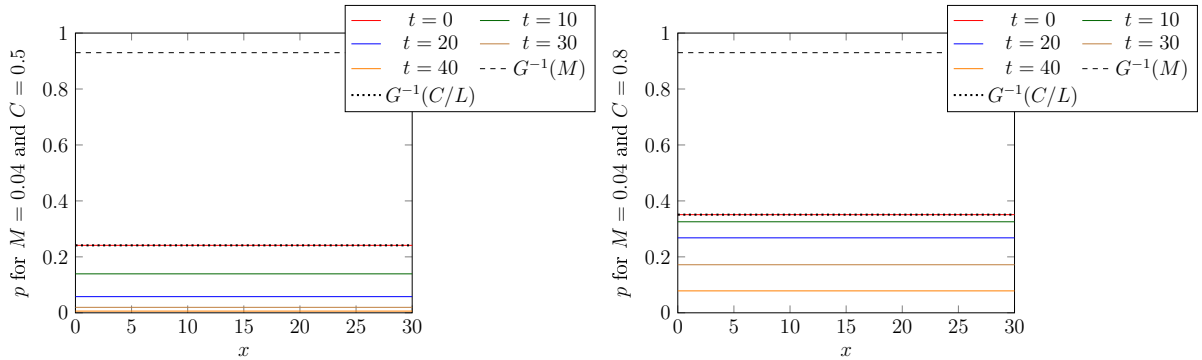

Figure 9.5: Case $C/|\Omega| < M$: Optimal solution p to Problem ($\mathcal{P}_{\text{reduced}}$) at time $t \in \{0, 10, 20, 30, 40\}$ for $C \in \{0.5, 0.8\}$ and $M = 0.04$ thanks to Uzawa algorithms

6 Perspectives

In a near future, we foresee to investigate a more involved model, closer to practical experiments, where one aims at determining release distributions in time and space, assuming that:

- releases are done periodically in time (for instance every week) and are impulses in time[4];
- at each release, the largest allowed amount of mosquitoes is released, corresponding to the maximal production capacity per week (which is relevant, according to the comparison principle).

As a consequence, we will be interested in determining the optimal way of releasing spatially the infected mosquitoes. Considering N releases, we denote by $t_0 = 0 < t_1 < \ldots < t_{N-1} < T$, $t_i = i\Delta T$, the release times. Rewriting the L^1 constraint on the control as $\langle u, 1\rangle_{\mathcal{D}',\mathcal{D}((0,T)\times\Omega)} \leqslant C$, the control function reads

$$u(t,x) = \sum_{i=0}^{N-1} u_i(x)\delta_{\{t=t_i\}}, \qquad \text{with } \sum_{i=0}^{N-1} \int_\Omega u_i(x)\,dx \leqslant C,$$

where the pointwise constraint is modified into $0 \leqslant u_i(\cdot) \leqslant M$.

The new optimal design problem reads

$$\inf_{\mathbf{u}\in\mathcal{V}_{C,M}} \tilde{J}_T(\mathbf{u}), \quad \text{where } \mathbf{u} = (u_i)_{0\leqslant i\leqslant N-1}, \quad \tilde{J}_T(\mathbf{u}) = J_T\left(\sum_{i=0}^{N-1} u_i(\cdot)\delta_{\{t=t_i\}}\right) \tag{$\mathcal{P}'_{\text{full}}$}$$

and

$$\mathcal{V}_{C,M} = \left\{\mathbf{u} = (u_i(\cdot))_{0\leqslant i\leqslant N-1},\ 0 \leqslant u_i \leqslant M \text{ a.e. in } \Omega,\ i \in \{0,\ldots,N-1\},\ \sum_{i=0}^{N-1}\int_\Omega u_i(x)\,dx \leqslant C\right\}.$$

[4] We consider Dirac measures since at the time-level of the study (namely, some generations), the release can be considered as instantaneous.

As done in this article, System (9.1) can be recast without source measure terms, coming from the specific form of the control functions.

In a second time, we will also look at dropping the assumption on the frequency of releases and determine optimal times of releases (in the spirit of [4], where a simpler ODE model were considered).

Another interesting question is also raised by the spatial heterogeneities. Indeed, in field experiments the environment is not homogeneous in space. Then an important issue, from an experimental point of view, is to determine how to adapt the releases with respect to the spacial heterogeneities to optimize the success of the replacement strategies.

Acknowledgments

M. Duprez, Y. Privat and N. Vauchelet were partially supported by the Project "Analysis and simulation of optimal shapes - application to lifesciences" of the Paris City Hall. Y. Privat was partially supported by the ANR Project ANR-18-CE40-0013 - SHAPO on Shape Optimization. We also warmly thank the referees for their comments and suggestions.

Chapitre 10

Conclusion

Dans cette thèse, nous avons d'abord rappelé dans le Chapitre 2 comment représenter un système de lois de conservation arbitraire avec un modèle de relaxation cinétique. Une méthode de splitting en temps avec sur-relaxation de paramètre ω s'avère particulièrement utile pour calculer numériquement les solutions de ce système.

Nous avons ensuite proposé dans les chapitres 3 et 4 plusieurs techniques mathématiques pour analyser la consistance et la stabilité des schémas de splitting. Nous avons construit une méthode pour calculer d'abord un système d'équations aux dérivées partielles équivalentes au schéma cinétique. Ce système porte sur les variables conservatives w et la variable d'écart au flux Y. Cette méthode générale a selon nous les originalités suivantes :

- Elle est complètement automatisable et l'équation peut être obtenue par un logiciel de calcul symbolique.
- Il est souvent préférable de grouper deux pas de temps pour l'analyse, afin de réduire les oscillations de l'écart au flux Y.
- Lorsque le paramètre de relaxation $\omega = 2$, le schéma de splitting est d'ordre 2, ce qui est bien connu. Nous avons montré que l'écart au flux n'a alors pas besoin d'être petit pour avoir une approximation consistante.
- Lorsque le paramètre de relaxation $\omega < 2$, le système équivalent contient un terme de relaxation raide en Y. Une analyse de Chapmann-Enskog permet dans un deuxième temps de retrouver la consistance du modèle cinétique avec le système diffusif classique sur w seul. Mais le système équivalent en (w, Y) donne des informations plus précises sur la stabilité, qui peuvent là aussi être obtenues automatiquement.
- Il semble que les informations de stabilité obtenues avec la technique du système d'équations équivalentes sont les mêmes que celles que l'on obtient avec l'analyse de stabilité avec l'entropie.

Comme perspective, il serait intéressant d'étendre l'analyse que nous avons réalisée pour l'équation de transport, à n'importe quel système hyperbolique. Il serait aussi intéressant d'automatiser les calculs d'équations équivalentes et de stabilité à tous les ordres.

Dans le Chapitre 5, nous avons tenté de réaliser une analyse double-échelle du système d'équations équivalentes. Cette étude est prometteuse, mais pour l'instant, pas assez pertinente pour modéliser correctement le comportement double-échelle du cas limite $\Delta t \to 0$. En perspective, il serait certainement intéressant de refaire l'analyse double-échelle avec un meilleur choix de fonction test pour mieux séparer les modes lents et rapides des solutions du système équivalent.

Dans le Chapitre 6, nous avons appliqué la théorie de la stabilité entropique pour construire des façons stables et précises d'appliquer les conditions aux limites pour l'équation de transport. La difficulté est que le nombre de conditions nécessaires est différent pour le système de lois de conservation et le système cinétique. Nous avons exploité l'analyse en entropie pour donner un critère suffisant de stabilité. Ce critère a été validé numériquement, mais les schémas obtenus retombent à l'ordre 1. Nous avons donc essayé de construire des conditions limites d'ordre 2. Certaines de ces conditions sont instables en temps long. Afin de rendre ces conditions limites d'ordre 2 stables, nous les avons projetées sur l'espace des conditions vérifiant le critère de décroissance de l'entropie. Nous obtenons alors des conditions limites numériquement stables et d'ordre 2.

La suite de la thèse a ensuite été consacrée à des applications des analyses précédentes.

Nous avons d'abord proposé dans le Chapitre 7 une première étude d'instabilité avec un modèle de drift en deux dimensions avec résolution de l'équation de transport par une méthode de Fourier. Puis, dans le chapitre 8, nous avons réalisé des simulations en trois dimensions du mouvement du plasma dans un tokamak. Plusieurs améliorations pourraient être effectuées pour rendre ces modélisations plus proches de la réalité. Nous avons considéré un modèle simplifié de drift-cinétique. Il serait intéressant d'utiliser des modèles plus complets pour modéliser le plasma, comme les systèmes de Vlasov-Poisson ou Vlasov-Maxwell. De plus, nous avons considéré des géométries relativement simples, avec plusieurs plans poloidaux alignés sur une droite, obtenant ainsi une géométrie cylindrique. L'ajout d'une courbure permettrait de modéliser un tore. Cela nécessite de prendre en compte des termes additionnels dans l'équation de transport. Il faudrait aussi pouvoir prendre en compte des maillages non structurés aussi dans la direction toroïdale.

Le Chapitre 9 est un travail indépendant du reste de la thèse. Dans ce travail nous analysons un modèle biologique de contrôle de population de moustiques.

Annexes

A Calcul des équations équivalentes

Dans cette annexe, nous rappelons le calcul des équations équivalentes sur w fait dans [34, 50]. Nous verrons dans l'Annexe B comment automatiser ces calculs à l'aide de Maple.

On considère le système d'équations cinétiques (2.3), que l'on peut écrire sous forme matricielle

$$\partial_t \boldsymbol{f} + \sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f} = \frac{1}{\varepsilon}(\boldsymbol{f}^{eq}(w) - \boldsymbol{f}), \tag{10.1}$$

où Λ_k sont les matrices diagonales définies par $(\Lambda_k)_{i,i} = \lambda_i^k$, pour tout $k = 1, \ldots, d$ et $i = 1, \ldots, n_v$. Les matrices Λ_k sont donc commutantes.

La solution w et les flux approximatifs z_k sont donnés par :

$$\begin{aligned} w(\boldsymbol{x}, t) &=< \mathbb{1}, \boldsymbol{f}(\boldsymbol{x}, t) >, \\ z_k(\boldsymbol{x}, t) &=< \mathbb{1}, \Lambda_k \boldsymbol{f}(\boldsymbol{x}, t) >, \qquad \text{pour } k = 1, \ldots, d. \end{aligned}$$

On considère que l'équation (10.1) est résolue par l'opérateur $\mathcal{M}^f$ défini par 2.15 (p.14) : on résout successivement une étape de transport, et une étape de relaxation.

L'étape de transport est donnée par

$$f_i^*(\boldsymbol{x}, t + \Delta t) = f_i(\boldsymbol{x} - \lambda_i \Delta t, t), \qquad \text{pour tout } i = 1, \ldots, n_v. \tag{10.2}$$

L'étape de relaxation est donnée par

$$\boldsymbol{f}(\boldsymbol{x}, t + \Delta t) = \boldsymbol{f}^*(\boldsymbol{x}, t + \Delta t) + \omega \left(\boldsymbol{f}^{eq}(w^*(\boldsymbol{x}, t + \Delta t)) - \boldsymbol{f}^*(\boldsymbol{x}, t + \Delta t) \right). \tag{10.3}$$

En effectuant un développement de Taylor de l'étape de transport (10.2) selon la variable t, on obtient

$$f^*(\boldsymbol{x}, t + \Delta t) = f(\boldsymbol{x}, t) - \Delta t \left(\sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x}, t) \right) + \frac{\Delta t^2}{2} \left(\sum_{k,l=1}^{d} \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}(\boldsymbol{x}, t) \right) + O(\Delta t^3). \tag{10.4}$$

En appliquant la fonction $< \mathbb{1}, . >$ à cette égalité, on obtient

$$\begin{aligned} w^*(\boldsymbol{x}, t+\Delta t) = w(\boldsymbol{x}, t) - \Delta t \left(\sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x}, t) \right) \\ + \frac{\Delta t^2}{2} \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}(\boldsymbol{x}, t) > + O(\Delta t^3). \end{aligned} \tag{10.5}$$

En appliquant la fonction $< \mathbb{1}, . >$ sur l'équation de relaxation (10.3), on obtient

$$\begin{aligned} w(\boldsymbol{x}, t+\Delta t) &= w^*(\boldsymbol{x}, t+\Delta t) + \omega \left(\sum_{i=1}^{n_v} f_i^{eq}(w^*(\boldsymbol{x}, t+\Delta t)) - w^*(\boldsymbol{x}, t+\Delta t) \right), \\ &= w^*(\boldsymbol{x}, t+\Delta t). \end{aligned}$$

Ce qui donne, en remplaçant w^* dans l'équation (10.5)

$$w(\boldsymbol{x}, t+\Delta t) = w(\boldsymbol{x}, t) - \Delta t \left(\sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x}, t) \right) + \frac{\Delta t^2}{2} \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}(\boldsymbol{x}, t) > + O(\Delta t^3). \tag{10.6}$$

En effectuant un développement de Taylor en t du terme de gauche, on obtient

$$\begin{aligned} w(\boldsymbol{x}, t) + \Delta t \partial_t w(\boldsymbol{x}, t) + \frac{\Delta t^2}{2} \partial_{tt} w(\boldsymbol{x}, t) \\ = w(\boldsymbol{x}, t) - \Delta t \left(\sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x}, t) \right) + \frac{\Delta t^2}{2} \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}(\boldsymbol{x}, t) > + O(\Delta t^3). \end{aligned} \tag{10.7}$$

En remplaçant $\boldsymbol{f}^*$ par l'expression (10.4) et $w(\boldsymbol{x}, t+\Delta t)$ par (10.6) dans l'étape de relaxation (10.3), on a

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}, t+\Delta t) = \boldsymbol{f}(\boldsymbol{x}, t) - \Delta t \left(\sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x}, t) \right) \\ + \omega \left[\boldsymbol{f}^{eq} \left(w(\boldsymbol{x}, t) - \Delta t \sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x}, t) \right) - \boldsymbol{f}(\boldsymbol{x}, t) + \Delta t \sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x}, t) \right] + O(\Delta t^2). \end{aligned}$$

En effectuant un développement de Taylor du membre gauche de l'équation précédente, on obtient

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}, t) + \Delta t \partial_t \boldsymbol{f}(\boldsymbol{x}, t) = \boldsymbol{f}(\boldsymbol{x}, t) - \Delta t \left(\sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x}, t) \right) \\ + \omega \left[\boldsymbol{f}^{eq} \left(w(\boldsymbol{x}, t) - \Delta t \sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x}, t) \right) - \boldsymbol{f}(\boldsymbol{x}, t) + \Delta t \sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x}, t) \right] + O(\Delta t^2). \end{aligned} \tag{10.8}$$

En appliquant la fonction $< \mathbb{1}, \Lambda_k . >$ à l'égalité (10.8), on obtient

$$\begin{aligned} z_j(\boldsymbol{x},t) + \Delta t \partial_t z_j(\boldsymbol{x},t) &= z_j(\boldsymbol{x},t) - \Delta t \left(\sum_{k=1}^{d} < \mathbb{1}, \Lambda_j \lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x},t) > \right) \\ &+ \omega \left[z_j^{eq} \left(w(\boldsymbol{x},t) - \Delta t \left(\sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x},t) \right) \right) - z_j(\boldsymbol{x},t) \right. \\ &\left. + \Delta t \left(\sum_{k=1}^{d} < \mathbb{1}, \Lambda_j \lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x},t) > \right) \right] + O(\Delta t^2). \end{aligned} \tag{10.9}$$

Finalement, les équations (10.7),(10.8) et (10.9) nous donne le système

$$\left\{ \begin{array}{ll} w(\boldsymbol{x},t) & +\Delta t \partial_t w(\boldsymbol{x},t) + \frac{\Delta t^2}{2} \partial_{tt} w(\boldsymbol{x},t) \\ & = w(\boldsymbol{x},t) - \Delta t \left(\sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x},t) \right) + \frac{\Delta t^2}{2} \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}(\boldsymbol{x},t) > \\ & +O(\Delta t^3). \\ \boldsymbol{f}(\boldsymbol{x},t) & +\Delta t \partial_t \boldsymbol{f}(\boldsymbol{x},t) = \boldsymbol{f}(\boldsymbol{x},t) - \Delta t \left(\sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x},t) \right) \\ & +\omega \left[\boldsymbol{f}^{eq} \left(w(\boldsymbol{x},t) - \Delta t \sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x},t) \right) - \boldsymbol{f}(\boldsymbol{x},t) + \Delta t \sum_{k=1}^{d} \Lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x},t) \right] \\ & +O(\Delta t^2). \\ z_j(\boldsymbol{x},t) & +\Delta t \partial_t z_j(\boldsymbol{x},t) = z_j(\boldsymbol{x},t) - \Delta t \left(\sum_{k=1}^{d} < \mathbb{1}, \Lambda_j \lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x},t) > \right) \\ & +\omega \left[z_j^{eq} \left(w(\boldsymbol{x},t) - \Delta t \left(\sum_{k=1}^{d} \partial_k z_k(\boldsymbol{x},t) \right) \right) - z_j(\boldsymbol{x},t) \right. \\ & \left. +\Delta t \left(\sum_{k=1}^{d} < \mathbb{1}, \Lambda_j \lambda_k \partial_k \boldsymbol{f}(\boldsymbol{x},t) > \right) \right] + O(\Delta t^2). \end{array} \right. \tag{10.10}$$

A.1 Linéarisation

Supposons que l'on puisse écrire

$$\left\{ \begin{array}{lllllll} w & = & w^0 & + & \Delta t w^1 & + & \Delta t^2 w^2 + O(\Delta t^3), \\ z_j & = & z_j^0 & + & \Delta t z_j^1 & + & O(\Delta t^2), \qquad \text{pour } j = 1, \ldots, d, \\ \boldsymbol{f} & = & \boldsymbol{f}^0 & + & \Delta t \boldsymbol{f}^1 & + & O(\Delta t^2). \end{array} \right. \tag{10.11}$$

Remplaçons w, z_j et $\boldsymbol{f}$ par les développements précédents dans le système (10.10) et regardons les termes obtenus pour chaque ordre.

Termes en $O(1)$

On ne conservant que les termes en $O(1)$, on obtient le système

$$\left\{ \begin{array}{lllll} w^0 & = & w^0, & & \\ z_j^0 & = & z_j^0 & + & \omega(z_j^{eq}(w^0) - z_j^0), \qquad \text{pour } j = 1, \ldots, d, \\ f^0 & = & f^0 & + & \omega(f^{eq}(w^0) - f^0). \end{array} \right.$$

On en déduit que

$$\left\{\begin{array}{lcl} z_j^0 & = & z_j^{eq}(w^0), \qquad\qquad \text{pour } j=1,\dots,d, \\ \boldsymbol{f}^0 & = & \boldsymbol{f}^{eq}(w^0). \end{array}\right. \tag{10.12}$$

Termes en $O(\Delta t)$

En ne conservant que les termes en $O(\Delta t)$, on obtient le système

$$\left\{\begin{array}{ll} w^1+\partial_t w^0 = & w^1-\sum_{k=1}^d \partial_k z_k^0, \\ \boldsymbol{f}^1+\partial_t \boldsymbol{f}^0 = & \boldsymbol{f}^1-\displaystyle\sum_{k=1}^d \Lambda_k\partial_k \boldsymbol{f}^0 \\ & +\omega\big[\partial \boldsymbol{f}^{eq}(w^0)\left(w^1-\displaystyle\sum_{k=1}^d \partial_k z_k^0\right)-\boldsymbol{f}^1+\displaystyle\sum_{k=1}^d \Lambda_k\partial_k\boldsymbol{f}^0\big], \\ z_j^1+\partial_t z_j^0 = & z_j^1-\sum_{k=1}^d <\mathbb{1},\Lambda_j\Lambda_k\partial_k\boldsymbol{f}^0> \\ & +\omega\big[\partial z_j^{eq}(w^0)\left(w^1-\sum_{k=1}^d\partial_k z_k^0\right)-z_j^1+\sum_{k=1}^d <\mathbb{1},\Lambda_j\Lambda_k\partial_k\boldsymbol{f}^0>\big]. \end{array}\right.$$

En utilisant les équations (10.12), on obtient

$$\left\{\begin{array}{ll} \partial_t w^0 = & -\sum_{k=1}^d \partial_k z_k^{eq}(w^0), \\ \partial_t \boldsymbol{f}^{eq}(w^0) = & (\omega-1)\left[\sum_{k=1}^d \Lambda_k\partial_k f^{eq}(w^0)\right] \\ & +\omega\big[\partial\boldsymbol{f}^{eq}(w^0)\left(w^1-\sum_{k=1}^d\partial_k z_k^{eq}(w^0)\right)-\boldsymbol{f}^1\big], \\ \partial_t z_j^{eq}(w^0) = & (\omega-1)\left[\sum_{k=1}^d <\mathbb{1},\Lambda_j\Lambda_k\partial_k\boldsymbol{f}^{eq}(w^0)>\right] \\ & +\omega\big[\partial z_j^{eq}(w^0)\left(w^1-\sum_{k=1}^d\partial_k z_k^{eq}(w^0)\right)-z_j^1\big]. \end{array}\right.$$

Ce qui donne

$$\left\{\begin{array}{ll} \partial_t w^0 = & -\sum_{k=1}^d \partial_k z_k^{eq}(w^0), \\ \boldsymbol{f}^1 = & \partial\boldsymbol{f}^{eq}(w^0)\left(w^1-\sum_{k=1}^d\partial_k z_k^{eq}(w^0)\right)-\frac{1}{\omega}\partial\boldsymbol{f}^{eq}(w^0)\partial_t(w^0) \\ & +(1-\frac{1}{\omega})\left[\sum_{k=1}^d\Lambda_k\boldsymbol{f}^{eq}(w^0)\right], \\ z_j^1 = & \partial z_j^{eq}(w^0)\left(w^1-\sum_{k=1}^d\partial_k z_k^{eq}(w^0)\right)-\frac{1}{\omega}\partial z_j^{eq}(w^0)\partial_t(w^0) \\ & +(1-\frac{1}{\omega})\left[\sum_{k=1}^d <\mathbb{1},\Lambda_j\Lambda_k\partial_k\boldsymbol{f}^{eq}(w^0)>\right]. \end{array}\right.$$

En remplaçant $\partial_t w^0$ par son expression dans la première équation du système dans les autres équations, on obtient

$$\left\{\begin{array}{ll} \partial_t w^0 & = -\sum_{k=1}^d \partial_k z_k^{eq}(w^0), \\ \boldsymbol{f}^1 & = \partial\boldsymbol{f}^{eq}(w^0)w^1+(1-\frac{1}{\omega})\Big[\partial\boldsymbol{f}^{eq}(w^0)\left(-\sum_{k=1}^d\partial_k z_k^{eq}(w^0)\right) \\ & \sum_{k=1}^d\Lambda_k\partial_k\boldsymbol{f}^{eq}(w^0)\Big], \\ z_j^1 & = \partial z_j^{eq}(w^0)w^1+(1-\frac{1}{\omega})\big[\partial z_j^{eq}(w^0)\left(-\sum_{k=1}^d\partial_k z_k^{eq}(w^0)\right) \\ & \sum_{k=1}^d <\mathbb{1},\Lambda_j\Lambda_k\partial_k\boldsymbol{f}^{eq}(w^0)>\big]. \end{array}\right. \tag{10.13}$$

Termes en $O(\Delta t^2)$

En ne conservant que les termes en $O(\Delta t^2)$, on obtient le système

$$w^2 + \partial_t w^1 + \frac{1}{2}\partial_{tt} w^0 = w^2 - \sum_{k=1}^{d} \partial_k z_k^1 + \frac{1}{2}\left(\sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^0 >\right).$$

Ce qui équivaut à

$$\partial_t w^1 = -\frac{1}{2}\partial_{tt} w^0 - \sum_{k=1}^{d} \partial_k z_k^1 + \frac{1}{2}\left(\sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^0 >\right). \tag{10.14}$$

De plus, en dérivant selon t la première équation du système (10.13), on a

$$\begin{aligned}\partial_{tt} w^0 &= -\partial_t \left[\sum_{k=1}^{d} \partial_k z_k^{eq}(w^0)\right], \\ &= -\sum_{k=1}^{d} \partial_k \left[\partial z_k^{eq}(w^0) \partial_t(w^0)\right].\end{aligned}$$

En remplaçant de nouveau $\partial_t w^0$ par son expression dans le système (10.13), on obtient

$$\partial_{tt} w^0 = \sum_{k=1}^{d} \partial_k \left[\partial z_k^{eq}(w^0)\left(\sum_{l=1}^{d} \partial_l z_l^{eq}(w^0)\right)\right]. \tag{10.15}$$

En dérivant selon x_j les équations en z_j^1 du système (10.13), on obtient, pour tout $j = 1, \ldots, d$

$$\begin{aligned}\partial_j z_j^1 =& \partial_j \left[\partial z_j^{eq}(w^0) w^1\right] + \left(\frac{1}{\omega} - 1\right) \partial_j \left[\partial z_j^{eq}(w^0)\left(\sum_{k=1}^{d} \partial_k z_k^{eq}(w^0)\right)\right] \\ &+ \left(1 - \frac{1}{\omega}\right)\left(\sum_{k=1}^{d} < \mathbb{1}, \Lambda_j \Lambda_k \partial_{k,l} \boldsymbol{f}^{eq}(w^0) >\right).\end{aligned} \tag{10.16}$$

En réinjectant les équations (10.15) et (10.16) dans l'équation (10.14), on obtient

$$\begin{aligned}\partial_t w^1 =& \left(\frac{1}{2} - \frac{1}{\omega}\right)\left(\sum_{k=1}^{d} \partial_k \left[\partial z_k^{eq}(w^0)\left(\sum_{l=1}^{d} \partial_l z_l^{eq}(w^0)\right)\right] - \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^{eq}(w^0) >\right) \\ &- \sum_{k=1}^{d} \partial_k \left[\partial z_k^{eq}(w^0) w^1\right].\end{aligned} \tag{10.17}$$

En additionnant cette équation avec la première équation du système (10.13) multipliée par

Δt, on obtient

$$\begin{aligned}
&\partial_t \left[w^0 + \Delta t w^1\right] + \sum_{k=1}^{d} \partial_k \left[z_k^{eq}(w^0) + \Delta t \partial z_k^{eq}(w^0) w^1\right] \\
&\quad = \Delta t \left(\frac{1}{2} - \frac{1}{\omega}\right) \left(\sum_{k=1}^{d} \partial_k \left[\partial z_k^{eq}(w^0) \left(\sum_{l=1}^{d} \partial_l z_l^{eq}(w^0)\right)\right] - \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^{eq}(w^0) >\right) \\
&\quad + O(\Delta t^2).
\end{aligned}$$

En utilisant (10.11), on obtient l'équation équivalente sur w

$$\begin{aligned}
&\partial_t w + \sum_{k=1}^{d} \partial_k q_k(w) \\
&\quad = \Delta t \left(\frac{1}{\omega} - \frac{1}{2}\right) \left(-\sum_{k=1}^{d} \partial_k \left[\partial q_k(w) \left(\sum_{l=1}^{d} \partial_l q_l(w)\right)\right] + \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^{eq}(w) >\right) \\
&\quad + O(\Delta t^2).
\end{aligned}$$

A.2 Applications

Dans cette sous-partie, nous appliquerons l'équation (A.1) aux différents modèles cinétiques présentés dans la section 2.

Modèle $D1Q2$

Pour le modèle $D1Q2$, défini dans la sous-section 2.1 (p.17), on a $d = 1$ et

$$\Lambda_1 = \begin{pmatrix} \lambda & 0 \\ 0 & -\lambda \end{pmatrix}.$$

En appliquant les paramètres de ce modèle à l'équation (A.1), on obtient

$$\partial_t w + \partial_x q(w) = \Delta t \left(\frac{1}{\omega} - \frac{1}{2}\right) \left(-\partial_x \left[q'(w) \partial_x q(w)\right] + \lambda^2 \partial_{xx} < \mathbb{1}, \boldsymbol{f}^{eq}(w) >\right).$$

En se rappelant que $< \mathbb{1}, \boldsymbol{f}^{eq}(w) >= w$, on obtient l'équation équivalente sur w du modèle $D1Q2$

$$\partial_t w + \partial_x q(w) = \Delta t \left(\frac{1}{\omega} - \frac{1}{2}\right) \left(\partial_x \left[\left(\lambda^2 - (q'(w))^2\right) \partial_x w\right]\right).$$

Modèle $D2Q3$

Pour le modèle $D2Q3$, défini dans la sous-section 2.2 (p.17), on a $d = 2$,

$$\Lambda_1 = \begin{pmatrix} \lambda & 0 & 0 \\ 0 & -\frac{\lambda}{2} & 0 \\ 0 & 0 & -\frac{\lambda}{2} \end{pmatrix} \qquad \text{et} \qquad \Lambda_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \frac{\lambda\sqrt{3}}{2} & 0 \\ 0 & 0 & -\frac{\lambda\sqrt{3}}{2} \end{pmatrix}.$$

En appliquant les paramètres de ce modèle à l'équation (A.1), on obtient

$$\begin{aligned}&\partial_t w + \nabla\cdot\boldsymbol{q}(w)\\&\quad= \Delta t\left(\frac{1}{\omega}-\frac{1}{2}\right)\left(-\sum_{k=1}^{d}\partial_k\left[q_k'(w)\left(\sum_{l=1}^{d}q_l'(w)\partial_l w\right)\right]+\sum_{k,l=1}^{d}<\mathbb{1},\Lambda_k\Lambda_l\partial_{k,l}\boldsymbol{f}^{eq}(w)>\right).\end{aligned}$$

En remplaçant Λ_1 et Λ_2 par leur expression, on obtient

$$\begin{aligned}&\sum_{k,l=1}^{d}<\mathbb{1},\Lambda_k\Lambda_l\partial_{k,l}\boldsymbol{f}^{eq}(w)>\\&\quad=\lambda^2\left(\partial_{1,1}\left[f_1^{eq}+\frac{1}{4}f_2^{eq}+\frac{1}{4}f_3^{eq}\right]+\frac{\sqrt{3}}{2}\partial_{1,2}\left[-f_2^{eq}+f_3^{eq}\right]+\frac{3}{4}\partial_{2,2}\left[f_2^{eq}+f_3^{eq}\right]\right).\end{aligned}$$

On rappelle que les vecteurs d'équilibre du modèle $D2Q3$ sont définis par (2.25)

$$\boldsymbol{f}_k^{eq}(\boldsymbol{w})=\frac{\boldsymbol{w}}{3}+\frac{2\boldsymbol{\lambda}_k\cdot\boldsymbol{q}(\boldsymbol{w})}{3\lambda^2}. \tag{10.18}$$

On obtient donc

$$\begin{aligned}&\sum_{k,l=1}^{d}<\mathbb{1},\Lambda_k\Lambda_l\partial_{k,l}\boldsymbol{f}^{eq}(w)>\\&\quad=\lambda^2\left(\partial_{1,1}\left[\frac{w}{2}+\frac{1}{2\lambda}q_1(w)\right]-\frac{1}{\lambda}\partial_{1,2}\left[q_2(w)\right]+\partial_{2,2}\left[\frac{w}{2}-\frac{1}{2\lambda}q_1(w)\right]\right),\\&\quad=\nabla\cdot\left(\begin{pmatrix}\frac{\lambda^2}{2}+\frac{\lambda}{2}q_1''(w) & -\frac{1}{2\lambda}q_2''(w)\\ -\frac{1}{2\lambda}q_2''(w) & \frac{\lambda^2}{2}-\frac{\lambda}{2}q_1''(w)\end{pmatrix}\nabla w\right).\end{aligned}$$

De plus, on peut écrire

$$-\sum_{k=1}^{d}\partial_k\left[q_k'(w)\left(\sum_{l=1}^{d}q_l'(w)\partial_l w\right)\right]=\nabla\cdot\left(\begin{pmatrix}-q_1'(w)^2 & -q_1'(w)q_2'(w)\\ -q_1'(w)q_2'(w) & -q_2'(w)^2\end{pmatrix}\nabla w\right).$$

Finalement, on obtient l'équation équivalente sur w du modèle $D2Q3$

$$\partial_t w+\nabla\cdot\boldsymbol{q}(w)=\Delta t\left(\frac{1}{\omega}-\frac{1}{2}\right)\nabla\cdot(\mathcal{D}_3\nabla w)+O(\Delta t^2), \tag{10.19}$$

avec la matrice de diffusion

$$\mathcal{D}_3=\begin{pmatrix}\frac{\lambda^2}{2}+\frac{\lambda}{2}q_1''(w)-q_1'(w)^2 & -\frac{1}{2\lambda}q_2''(w)-q_1'(w)q_2'(w)\\ -\frac{1}{2\lambda}q_2''(w)-q_1'(w)q_2'(w) & \frac{\lambda^2}{2}-\frac{\lambda}{2}q_1''(w)-q_2'(w)^2\end{pmatrix}.$$

Modèle $D2Q4$

Pour le modèle $D2Q4$, défini dans la sous-section 2.3 (p.18), on a $d = 2$,

$$\Lambda_1 = \begin{pmatrix} \lambda & 0 & 0 & 0 \\ 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad \text{et} \qquad \Lambda_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & -\lambda \end{pmatrix}.$$

En appliquant les paramètres de ce modèle à l'équation (A.1), on obtient

$$\begin{aligned} &\partial_t w + \nabla \cdot \boldsymbol{q}(w) \\ &\quad = \Delta t \left(\frac{1}{\omega} - \frac{1}{2}\right) \left(-\sum_{k=1}^{d} \partial_k \left[q_k'(w) \left(\sum_{l=1}^{d} q_l'(w) \partial_l w \right) \right] + \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^{eq}(w) > \right). \end{aligned}$$

En remplaçant Λ_1 et Λ_2 par leur expression, on obtient

$$\begin{aligned} \sum_{k,l=1}^{d} < \mathbb{1}, \Lambda_k \Lambda_l \partial_{k,l} \boldsymbol{f}^{eq}(w) > &= \lambda^2 \left(\partial_{1,1} \left[f_1^{eq}(w) + f_2^{eq}(w) \right] + \partial_{2,2} \left[f_3^{eq}(w) + f_4^{eq}(w) \right] \right), \\ &= \frac{\lambda^2}{2} \left(\partial_{1,1} w + \partial_{2,2} w \right), \end{aligned}$$

car le vecteur d'équilibre $\boldsymbol{f}_k^{eq}(\boldsymbol{w})$ du modèle $D2Q4$ est défini par (2.27)

$$f_k^{eq}(w) = \frac{w}{4} + \frac{\boldsymbol{\lambda}_k \cdot \boldsymbol{q}(w)}{2\lambda^2}.$$

Finalement, on obtient l'équation équivalente sur w du modèle $D2Q4$

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = \Delta t \left(\frac{1}{\omega} - \frac{1}{2}\right) \nabla \cdot (\mathcal{D}_4 \nabla w) + O(\Delta t^2), \tag{10.20}$$

avec la matrice de diffusion

$$\mathcal{D}_4 = \begin{pmatrix} \frac{\lambda^2}{2} - q_1'(w)^2 & -q_1'(w) q_2'(w) \\ -q_1'(w) q_2'(w) & \frac{\lambda^2}{2} - q_2'(w)^2 \end{pmatrix}.$$

B Calcul des systèmes équivalents et équations équivalentes avec Maple

Cette annexe présente le code Maple. Dans la section B.1, les différents modèles cinétiques utilisés ($D1Q2$, $D2Q3$ et $D2Q4$) sont définis. Attention, lorsque le bloc de code commence par ###, il ne faut l'exécuter que si elle correspond au modèle cinétique choisi. Dans la section B.2, les opérateurs de transport et de relaxation sont définis, ainsi que des fonctions qui seront utiles pour séparer les différentes dérivées. Dans la section B.3, le système équivalent en (w, Y) au modèle cinétique est calculé, il est ensuite simplifié en une équation équivalente en w dans la section B.4. La section B.5 permet de calculer la matrice $P2$ permettant de diagonaliser le système équivalent en (w, Y). Cette matrice devant être définie positive, on obtient des conditions d'hyperbolicité sur le système. Enfin, dans la section B.6, on effectue un changement de variables dans les inconnues cinétiques f_i sur la matrice $P2$ afin d'obtenir l'expression de l'entropie dans ces variables.

B.1 Définition des différents modèles cinétiques

```
> restart : with(linalg) : with(LinearAlgebra) :
```

```
> ### Pour le modèle D1Q2
nv := 2;
d := 1;
x := (x1);
W := vector(nv, [w(x1), y1(x1)]);
Wp := vector(nv, [w, y1]);
v1 := vector(2, [lambda]);
v2 := vector(2, [−lambda]);
s := (vi, v, dt) → subs(x1 = x1 − vi[1] ∗ dt, v);
Delta := (V, dt) → [s(v1, V[1], dt), s(v2, V[2], dt)];
Q := (V, a) → [V[1], V[2] + a ∗ q1(V[1])];
M := matrix(nv, nv, [1, 1, lambda, −lambda]);
```

```
> ### Pour le modèle D2Q3
nv := 3;
d := 2;
x := (x1, x2);
W := vector(nv, [w(x1, x2), y1(x1, x2), y2(x1, x2)]);
Wp := vector(nv, [w, y1, y2]);
v1 := vector(2, [lambda, 0]);
v2 := vector(2, [−(1/2) ∗ lambda, (1/2) ∗ lambda ∗ sqrt(3)]);
v3 := vector(2, [−(1/2) ∗ lambda, −(1/2) ∗ lambda ∗ sqrt(3)]);
s := (vi, v, dt) → subs(x1 = x1 − vi[1] ∗ dt, x2 = x2 − vi[2] ∗ dt, v);
Delta := (V, dt) → [s(v1, V[1], dt), s(v2, V[2], dt), s(v3, V[3], dt)];
Q := (V, a) → [V[1], V[2] + a ∗ q1(V[1]), V[3] + a ∗ q2(V[1])];
M := matrix(nv, nv, [1, 1, 1,
```

```
        lambda, −(1/2) ∗ lambda, −(1/2) ∗ lambda,
        0, (1/2) ∗ lambda ∗ sqrt(3), −(1/2) ∗ lambda ∗ sqrt(3)]);

> ### Pour le modèle D2Q4
nv := 4;
d := 2;
x := (x1, x2);
W := vector(nv, [w(x1, x2), y1(x1, x2), y2(x1, x2), z3(x1, x2)]);
Wp := vector(nv, [w, y1, y2, z3]);
v1 := vector(2, [lambda, 0]);
v2 := vector(2, [−lambda, 0]);
v3 := vector(2, [0, lambda]);
v4 := vector(2, [0, −lambda]);
s := (vi, v, dt) → subs(x1 = x1 − vi[1] ∗ dt, x2 = x2 − vi[2] ∗ dt, v);
Delta := (V, dt) → [s(v1, V[1], dt), s(v2, V[2], dt), s(v3, V[3], dt), s(v4, V[4], dt)];
Q := (V, a) → [V[1], V[2] + a ∗ q1(V[1]), V[3] + a ∗ q2(V[1]), V[4]];
M := matrix(nv, nv, [1, 1, 1, 1,
        lambda, −lambda, 0, 0,
        0, 0, lambda, −lambda,
        lambda^2, lambda^2, −lambda^2, −lambda^2]);
```

B.2 Définition des opérateurs et autres fonctions

```
> # Si i=1, alors on retourne a, sinon, on retourne b
ind :=proc(i, a, b)
local vv :
if i = 1 then
    vv :=a :
else
    vv :=b :
end if :
return (vv) :
end proc :


> # Opérateur de relaxation
R :=(W, omega) → vector(nv, (k) → ind(k, 1, 1 − omega) · W[k]) :


> # Opérateur de transport sur (w,Y)
T := (V, dt) → evalm(M& ∗ Delta(evalm(inverse(M)& ∗ V), dt)) :
T2 := (V, dt) → Q(T(Q(V, 1), dt), −1) :


> # Développement en séries
```

```
dev :=(pde) → vector(nv, (i) → −expand(series(pde[i], dt = 0, 3))):


> # Approximation en différences finies de D_t(W)
Gamma2 :=(L2, L1, W, L_i, L_i2) → (L1 − L_i)/2:
Gamma3 :=(L2, L1, W, L_i, L_i2) → (−L2 + 8 · L1 − 8 · L_i + L_i2)/12:
pde :=(Gamma, dt, L2, L1, W, L_i, L_i2) →
    vector(nv, (k) → Gamma(L2[k], L1[k], W[k], L_i[k], L_i2[k])/dt):


> # Supprime les dérivées et d érivées secondes de y dans v
suppr1 :=(y, v) → simplify(subs([D_1(y)(x) = 0, D_2(y)(x) = 0], v)):
suppr2 :=(y, v) →
    simplify(subs([D_{1,1}(y)(x) = 0, D_{1,2}(y)(x) = 0, D_{2,2}(y)(x) = 0], v)):
suppr :=(y, v) → suppr1(y, suppr2(y, v)):
fs :=(v) → factor(simplify(v)):


> # Supprime les dérivées et dérivées secondes de y dans v
suppr1 :=proc(y, v)
local vv, k:
vv :=v:
for k from 1 to d do
    vv :=simplify(subs(D_k(y)(x) = 0, vv)):
end do:
return (vv):
end proc:
suppr2 :=proc(y, v)
local vv, i, j:
vv :=v:
for i from 1 to d do
    for j from 1 to d do
        vv :=simplify(subs(D_{i,j}(y)(x) = 0, vv)):
    end do:
end do:
return (vv):
end proc:

suppr :=(y, v) → suppr1(y, suppr2(y, v)):

fs :=(v) → factor(simplify(v)):


> # Conserve dans v uniquement les dérivées et dérivées secondes de y
```

```
s1_y :=proc(v, y)
local x, vv, i:
vv :=v:
for i from 1 to nv do
    if y ≠ Wp[i] then
        vv :=suppr(Wp[i], vv)
    end if:
end do:
return (vv):
end proc:
```

```
> # Conserve dans v uniquement les dérivées selon x_ i
s1_i :=proc(v, i)
local ii, j, k, vv:
vv :=v:
for ii from 1 to d do
    ifii ≠ i then
        for k from 1 to nv do
            vv :=simplify(subs(D_ii(Wp[k])(x) = 0, vv))
        end do:
    end if:
end do:
return vv;
end proc:
```

```
> # Supprime toutes les dérivées secondes, sauf D_{i,j}
s2_ij :=proc(v, i, j)
local ii, jj, k, L, m, vv:
vv :=v:
for ii from 1 to d do
    for jj from 1 to d do
        if(ii ≠i or jj ≠ j) and (i = j or ii ≠ j or jj ≠ i) then
            for m from 1 to nv do
                vv :=simplify(subs(D_{ii,jj}(Wp[m])(x) = 0, vv)):
            end do:
        end if:
    end do:
end do:
return vv;
end proc:
```

B.3 Système équivalent en (w, Y)

```
> # Opérateur symétrique S
```

$\mathrm{S1} := \mathrm{T2}(\mathrm{R}(\mathrm{T2}(\mathrm{T2}(\mathrm{R}(\mathrm{T2}(\mathrm{W}, \mathrm{dt}/4), \mathrm{omega}), \mathrm{dt}/4), \mathrm{dt}/4), \mathrm{omega}), \mathrm{dt}/4):$

$\mathrm{S_i} := \mathrm{subs}(\{\mathrm{dt} = -\mathrm{dt}, \mathrm{omega} = 1 - 1/(1 - \mathrm{omega})\}, \mathrm{S1}):$

> # Système équivalent en $O(Dt^2)$

$\mathrm{pdeS1} := \mathrm{pde}(\mathrm{Gamma2}, \mathrm{dt}, 0, \mathrm{S1}, 0, \mathrm{S_i}, 0):$

$\mathrm{eS} := \mathrm{dev}(\mathrm{pdeS1}):$

```
> # Pour un flux linéaire
```

$\mathrm{q1} := (\mathrm{w}) \rightarrow \mathrm{a} \cdot \mathrm{w}:$

$\mathrm{q2} := (\mathrm{w}) \rightarrow \mathrm{b} \cdot \mathrm{w}:$

$\mathrm{v} := [\mathrm{a}, \mathrm{b}]:$

> # Termes en $O(1/Dt)$

$$\mathrm{C0} := (\mathrm{e}) \rightarrow \mathrm{Matrix}\left(\mathrm{nv}, 1, (\mathrm{i}, \mathrm{j}) \rightarrow \mathrm{fs}\left(\frac{\mathrm{coeff}(\mathrm{e[i]}, \mathrm{dt}, -1)}{(\mathrm{Wp[i]})(\mathrm{x})}\right)\right):$$

$\mathrm{A} := \mathrm{C0}(\mathrm{eS});$

> # Termes en $O(1)$

$$\mathrm{C1} := (\mathrm{e}, \mathrm{k}) \rightarrow \mathrm{Matrix}(\mathrm{nv}, \mathrm{nv}, (\mathrm{i}, \mathrm{j}) \rightarrow \mathrm{fs}(\frac{\mathrm{s1_y}(\mathrm{s1_i}(\mathrm{coeff}(\mathrm{e[i]}, \mathrm{dt}, 0), \mathrm{k}), \mathrm{Wp[j]})}{\mathrm{D_k}(\mathrm{Wp[j]})(\mathrm{x})})):$$

$\mathrm{B1} := \mathrm{C1}(\mathrm{eS}, 1);$

```
> ### Si d=2
```

$\mathrm{B2} := \mathrm{C1}(\mathrm{eS}, 2);$

B.4 Simplification du système équivalent en (w, Y) en une équation équivalente en w

```
> # Expression de y selon les dérivées spatiales de w
```

$$\begin{aligned}\mathrm{y_Dt} := (\mathrm{i}) \rightarrow \mathrm{fs}((&-\mathrm{s1_y}(\mathrm{s1_i}(\mathrm{coeff}(\mathrm{eS[i+1]}, \mathrm{dt}, 0), 1), \mathrm{Wp[1]}) \\ &-\mathrm{s1_y}(\mathrm{s1_i}(\mathrm{coeff}(\mathrm{eS[i+1]}, \mathrm{dt}, 0), 2), \mathrm{Wp[1]})) \\ &/(\mathrm{coeff}(\mathrm{eS[i+1]}, \mathrm{dt}, -1)) \cdot \mathrm{dt} \cdot \mathrm{Wp[i+1]}(\mathrm{x})):\end{aligned}$$

```
> ### Si d=1
W_G :=vector(nv, [w(x), y_Dt(1)]):
y_Dt(1);
```

```
> ### Si d=2
W_G :=vector(nv, [w(x), y_Dt(1), y_Dt(2)]):
y_Dt(1);
y_Dt(2);
```

```
> S1_G :=T2(R(T2(T2(R(T2(W_G, dt/4), omega), dt/4), dt/4), omega), dt/4):
S_i_G :=subs({dt = −dt, omega = 1 − 1/(1 − omega)}, S1_G):
pdeS1_G :=pde(Gamma2, dt, 0, S1_G, 0, S_i_G, 0):
eS_G :=dev(pdeS1_G):
```

```
> # Coefficient en O(1/Dt)
coeff(eS_G[1], dt, −1);
```

```
> # Coefficient en O(1)
coeff(eS_G[1], dt, 0);
```

```
> # Coefficient en O(Dt)
```

$$\mathrm{Matrix}\left(\mathrm{d}, \mathrm{d}, (\mathrm{i}, \mathrm{j}) \rightarrow \mathrm{fs}\left(\frac{\mathrm{s2_ij}(\mathrm{coeff}(\mathrm{eS_G}[1], \mathrm{dt}, 1), \mathrm{i}, \mathrm{j})}{\mathrm{D_{i,j}}(\mathrm{w})(\mathrm{x})}\right)\right);$$

B.5 Condition d'hyperbolicité

```
> ### Pour le modèle D1Q2
P :=matrix(2, 2, [1, 0, 0, p3]):
M1 :=evalm(P &∗ B1):
sol :=solve({M1[1, 2] = M1[2, 1]}, {p1, p2, p3}):
P :=factor~(evalm(subs(sol, eval(P)))):
P2 :=simplify~(evalm(subs~(p2 = 0, p1 = 1, P)));
```

```
> ### Pour le modèle D2Q3
P :=matrix(nv, nv, [p1, p2, p3, p2, p4, p5, p3, p5, p6]):
M1 :=evalm(P &∗ B1): M2 :=evalm(P &∗ B2):
```

```
sol :=solve({M1[1,2] = M1[2,1], M1[1,3] = M1[3,1], M1[2,3] = M1[3,2], M2[1,2] = M2[2,1], M2[1,3] =
M2[3,1], M2[2,3] = M2[3,2]}, {p1, p2, p3, p4, p5, p6}):
P :=factor~(evalm(subs(sol, eval(P)))):
P2 :=simplify~(evalm(subs~(p5 = b · (2a + lambda), P)));
```

```
> ### Pour le modèle D2Q4
P :=matrix(4, 4, [p1, p2, p3, p4, p2, p5, p6, p7, p3, p6, p8, p9, p4, p7, p9, p10]):
M1 :=evalm(P &∗ B1): M2 :=evalm(P &∗ B2):
sol :=solve({M1[1,2] = M1[2,1], M1[1,3] = M1[3,1], M1[2,3] = M1[3,2], M1[1,4] = M1[4,1], M1[2,4] =
M1[4,2], M1[4,3] = M1[4,3], M2[1,2] = M2[2,1], M2[1,3] = M2[3,1], M2[2,3] = M2[3,2], M2[1,4] =
M2[4,1], M2[2,4] = M2[4,2], M2[4,3] = M2[4,3]}, {p1, p2, p3, p4, p5, p6, p7, p8, p9, p10}):
P :=factor~(evalm(subs(sol, eval(P)))):
P2 :=simplify~(evalm(subs~(p7 = 2 · a · (2b − lambda) · (2 · b + lambda), P)));
```

B.6 Matrice d'entropie

```
> # Pour obtenir la matrice de passage entre F et (w,Y)
Mz_to_My :=proc(M)
local i, j, B:
B :=matrix(nv, nv);
for i from 1 to nv do
    for j from 1 to nv do
        if i > 1 and i ≤ 1 + d then
            B[i, j]:= M[i, j] − v[i − 1];
        else
            B[i, j]:= M[i, j];
        end if:
    end do:
end do:
return B:
end proc:
```

```
> My :=Mz_to_My(M):
evalm(My);
```

```
> # Matrice de l'entropie
S :=fs~(evalm(transpose(My) &∗ P2 &∗ My))
```

Bibliographie

[1] Grégoire Allaire. Homogenization and two-scale convergence. *SIAM Journal on Mathematical Analysis*, 23(6) :1482–1518, 1992.

[2] Luis Almeida, Michel Duprez, Yannick Privat, and Nicolas Vauchelet. Mosquito population control strategies for fighting against arboviruses. *Mathematical Biosciences and Engineering*, 16(6) :6274, 2019.

[3] Luis Almeida, Antoine Haddon, Claire Kermorvant, Alexis Léculier, Yannick Privat, Martin Strugarek, Nicolas Vauchelet, and Jorge P. Zubelli. Optimal release of mosquitoes to control dengue transmission. *ESAIM : ProcS*, 67 :16–29, 2020.

[4] Luis Almeida, Yannick Privat, Martin Strugarek, and Nicolas Vauchelet. Optimal releases for population replacement strategies : application to *Wolbachia*. *SIAM J. Math. Anal.*, 51(4) :3170–3194, 2019.

[5] Denise Aregba-Driollet and Roberto Natalini. Discrete kinetic schemes for systems of conservation laws. In Michael Fey and Rolf Jeltsch, editors, *Hyperbolic Problems : Theory, Numerics, Applications*, pages 1–10, Basel, 1999. Birkhäuser Basel.

[6] Denise Aregba-Driollet and Roberto Natalini. Discrete kinetic schemes for multidimensional systems of conservation laws. *SIAM Journal on Numerical Analysis*, 37(6) :1973–2004, 2000.

[7] Jayesh Badwaik, Matthieu Boileau, David Coulette, Emmanuel Franck, Philippe Helluy, Christian Klingenberg, Laura Mendoza, and Herbert Oberlin. Task-based parallelization of an implicit kinetic scheme. *ESAIM : Proceedings and Surveys*, 63 :60–77, 2018.

[8] N. H. Barton and Michael Turelli. Spatial waves of advance with bistable dynamics : cytoplasmic and genetic analogues of Allee effects. *The American Naturalist*, 178 :E48–E75, 2011.

[9] Logan Beal, Daniel Hill, R. Martin, and John Hedengren. GEKKO optimization suite. *Processes*, 6(8) :106, 2018.

[10] Thomas Bellotti. Truncation errors and modified equations for the lattice Boltzmann method via the corresponding finite difference schemes. *ESAIM : M2AN*, 57(3) :1225–1255, 2023.

[11] Nicolas Besse and Michel Mehrenberger. Convergence of classes of high-order semi-Lagrangian schemes for the Vlasov-Poisson system. *Mathematics of computation*, 77(261) :93–123, 2008.

[12] Pierre-Alexandre Bliman. Feedback control principles for biological control of dengue vectors. *18th European Control Conference (ECC), arXiv preprint arXiv :1903.00730*, 2019.

[13] François Bouchut. Construction of BGK models with a family of kinetic entropies for a given system of conservation laws. *Journal of Statistical Physics*, 95(1-2) :113–170, 1999.

[14] François Bouchut. Stability of relaxation models for conservation laws. In *European Congress of Mathematics*, pages 95–101. Eur. Math. Soc., 2005.

[15] Françoise Bourdel, Philippe Delorme, and Pierre-Alain Mazet. Convexity in hyperbolic problems. Application to a discontinuous Galerkin method for the resolution of the polydimensional Euler equations. In *Nonlinear Hyperbolic Equations—Theory, Computation Methods, and Applications : Proceedings of the Second International Conference on Nonlinear Hyperbolic Problems, Aachen, FRG, March 14 to 18, 1988*, pages 31–42. Springer, 1989.

[16] Françoise Bourdel, Pierre-Alain Mazet, Jean-Pierre Croisille, and Philippe Delorme. On the approximation of K-diagonalizable hyperbolic systems by finite elements-Applications to the Euler equations and to gaseous mixtures. *La Recherche Aerospatiale (English Edition)(ISSN 0379-380X*, 5 :15–34, 1989.

[17] Kostas Bourtzis. *Wolbachia*-based technologies for insect pest population control. In *Transgenesis and the management of vector-borne disease*, pages 104–113. Springer, 2008.

[18] Bérenger Bramas and Alain Ketterlin. Improving parallel executions by increasing task granularity in task-based runtime systems using acyclic DAG clustering. *PeerJ Computer Science*, 6 :e247, 2020.

[19] Doris E. Campo-Duarte, Olga Vasilieva, Daiver Cardona-Salgado, and Mikhail Svinin. Optimal control approach for establishing *wMelPop Wolbachia* infection among wild *Aedes aegypti* populations. *J. Math. Biol.*, 76(7) :1907–1950, 2018.

[20] Shiyi Chen and Gary D. Doolen. Lattice Boltzmann method for fluid flows. *Annual Review of Fluid Mechanics*, 30(1) :329–364, 1998.

[21] Edward D. Conway and Joel A. Smoller. A comparison technique for systems of reaction-diffusion equations. *Communications in Partial Differential Equations*, 2(7) :679–697, 1977.

[22] David Coulette, Emmanuel Franck, Philippe Helluy, Michel Mehrenberger, and Laurent Navoret. High-order implicit palindromic Discontinuous Galerkin method for kinetic-relaxation approximation. *Computers & Fluids*, 2019.

[23] Jean-François Coulombel and Frédéric Lagoutière. The Neumann numerical boundary condition for transport equations. *Kinetic and Related Models* , 13(1) :1–32, February 2020.

[24] Clémentine Courtès, David Coulette, Emmanuel Franck, and Laurent Navoret. Vectorial kinetic relaxation model with central velocity. Application to implicit relaxations schemes. *Communications in Computational Physics*, 27(4), 2020.

[25] Jean-Pierre Croisille. *Contribution à l'étude théorique et à l'approximation par éléments finis du système hyperbolique de la dynamique des gaz multidimensionnelle et multiespèces*. PhD thesis, Paris 6, 1990.

[26] Nicolas Crouseilles, Pierre Glanc, Sever Adrian Hirstoaga, Eric Madaule, Michel Mehrenberger, and Jérôme Pétri. A new fully two-dimensional conservative semi-Lagrangian method : applications on polar grids, from diocotron instability to ITG turbulence. *The European Physical Journal D : Atomic, molecular, optical and plasma physics*, 68(9) :DOI : 10.1140/epjd/e2014–50180–9, September 2014. article 252.

[27] Nicolas Crouseilles, Michel Mehrenberger, and Eric Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *J. Comput. Phys.*, 229(6) :1927–1953, 2010.

[28] Paul J. Dellar. An interpretation and derivation of the lattice Boltzmann method using Strang splitting. *Computers & Mathematics with Applications*, 65(2) :129–141, 2013.

[29] Suresh M. Deshpande. A second-order accurate kinetic-theory-based method for inviscid compressible flows. Technical report, 1986.

[30] Firas Dhaouadi, Emilie Duval, Sergey Tkachenko, and Jean-Paul Vila. Stability theory for some scalar finite difference schemes : validity of the modified equations approach. *ESAIM : Proceedings and Surveys*, 70 :124–136, 2021.

[31] Giacomo Dimarco and Raphaël Loubere. Towards an ultra efficient kinetic scheme. Part I : Basics on the BGK equation. *J. Comput. Physics*, 255 :680–698, 2013.

[32] Giacomo Dimarco, Raphaël Loubère, Jacek Narski, and Thomas Rey. An efficient numerical method for solving the Boltzmann equation in multidimensions. *J. Comput. Phys.*, 353 :46–81, 2018.

[33] Florence Drui, Emmanuel Franck, Philippe Helluy, and Laurent Navoret. An analysis of over-relaxation in a kinetic approximation of systems of conservation laws. *Comptes Rendus Mécanique*, 347(3) :259–269, January 2019.

[34] François Dubois. Equivalent partial differential equations of a lattice Boltzmann scheme. *Computers & Mathematics with Applications*, 55(7) :1441–1449, 2008.

[35] François Dubois. Simulation of strong nonlinear waves with vectorial lattice Boltzmann schemes. *International Journal of Modern Physics C*, 25(12) :1441014, 2014.

[36] François Dubois. Nonlinear fourth order Taylor expansion of lattice Boltzmann schemes. *Asymptotic Analysis*, (Preprint) :1–41, 2021.

[37] François Dubois, Tony Février, and Benjamin Graille. Lattice Boltzmann schemes with relative velocities. *Communications in Computational Physics*, 17(4) :1088–1112, 2015.

[38] François Dubois and Pierre Lallemand. Towards higher order lattice Boltzmann schemes. *Journal of Statistical Mechanics*, 2009(P06) :P06006, June 2009.

[39] François Dubois and Philippe Le Floch. Boundary conditions for nonlinear hyperbolic systems of conservation laws. *Journal of Differential Equations*, 71(1) :93–122, 1988.

[40] Geverton Leandro Carneiro Dutra, Lilha Maria Barbosa dos Santos, Eric Pearce Caragata, Jéssica Barreto Lopes Silva, Daniel A. M. Villela, Rafael Maciel-de Freitas, and Luciano Andrade Moreira. From lab to field : the influence of urban landscapes on the invasive potential of *Wolbachia* in brazilian *Aedes aegypti* mosquitoes. *PLoS Negl Trop Dis*, 9(4), 2015.

[41] Victor A. Dyck, Jorge Hendrichs, and A.S. Robinson. *Sterile insect technique : principles and practice in area-wide integrated pest management*. Springer, 2006.

[42] Lawrence C. Evans. *Partial differential equations. 2nd ed*, volume 19. Providence, RI : American Mathematical Society (AMS), 2nd ed. edition, 2010.

[43] József Z. Farkas and Peter Hinow. Structured and unstructured continuous models for *Wolbachia* infections. *Bull. Math. Biol.*, 72(8) :2067–2088, 2010.

[44] Andrew Fenton, Karyn N. Johnson, Jeremy C. Brownlie, and Gregory D.D. Hurst. Solving the *Wolbachia* paradox : modeling the tripartite interaction between host, *Wolbachia*, and a natural enemy. *The American Naturalist*, 178(3) :333–342, 2011.

[45] Tony Février. *Extension et analyse des schémas de Boltzmann sur réseau : les schémas à vitesse relative*. Theses, Université Paris Sud - Paris XI, December 2014.

[46] Dana A. Focks, Dan G. Haile, Eric Daniels, and Gary A. Mount. Dynamic life table model for Aedes aegypti (Diptera : Culicidae) : analysis of the literature and model development. *Journal of Medical Entomology*, 30(6) :1003–1017, 11 1993.

[47] Emmanuel Frenod. Two-scale convergence. *ESAIM : Proceedings*, 38 :1–35, December 2012.

[48] Guoliang Fu, Rosemary S. Lees, Derric Nimmo, Diane Aw, Li Jin, Pam Gray, Thomas U. Berendonk, Helen White-Cooper, Sarah Scaife, Hoang Kim Phuc, et al. Female-specific flightless phenotype for mosquito control. *Proceedings of the National Academy of Sciences*, 107(10) :4550–4554, 2010.

[49] Radek Fucik and Robert Straka. Equivalent finite difference and partial differential equations for the lattice Boltzmann method. *Computers & Mathematics with Applications*, 90 :96–103, 2021.

[50] Benjamin Graille. Approximation of mono-dimensional hyperbolic systems : a lattice Boltzmann scheme as a relaxation method. *Journal of Computational Physics*, 266 :74–88, 2014.

[51] Virginie Grandgirard, Jérémie Abiteboul, Julien Bigot, Thomas Cartier-Michaud, Nicolas Crouseilles, Guilhem Dif-Pradalier, Charles Ehrlacher, Damien Esteve, Xavier Garbet, Philippe Ghendrih, et al. A 5d gyrokinetic full-f global semi-Lagrangian code for flux-driven ion turbulence simulations. *Computer Physics Communications*, 207 :35–68, 2016.

[52] Hervé Guillard, Jalal Lakhlili, Adrien Loseille, Alexis Loyer, Boniface Nkonga, Ahmed Ratnani, and Ali Elarif. Tokamesh : a software for mesh generation in Tokamaks. Research Report, RR-9230, CASTOR., 2018.

[53] Amiram Harten. On the symmetric form of systems of conservation laws with entropy. *Journal of computational physics*, 49, 1983.

[54] Amiram Harten, Peter D. Lax, and Bram van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM review*, 25(1) :35–61, 1983.

[55] Jörg C. Heinrich and Maxwell J. Scott. A repressible female-specific lethal genetic system for making transgenic insect strains suitable for a sterile-release program. *Proceedings of the National Academy of Sciences*, 97(15) :8229–8232, 2000.

[56] Jan S. Hesthaven and Tim Warburton. *Nodal discontinuous Galerkin methods : algorithms, analysis, and applications*. Springer Science & Business Media, 2007.

[57] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex analysis and minimization algorithms. I*, volume 305 of *Grundlehren der Mathematischen Wissenschaften [Fundamental Principles of Mathematical Sciences]*. Springer-Verlag, Berlin, 1993. Fundamentals.

[58] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2004.

[59] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex analysis and minimization algorithms I : Fundamentals*, volume 305. Springer science & business media, 2013.

[60] Harriet Hughes and Nicholas F. Britton. Modelling the use of *Wolbachia* to control dengue fever transmission. *Bull. Math. Biol.*, 75(5) :796–818, 2013.

[61] Shi Jin and Zhouping Xin. The relaxation schemes for systems of conservation laws in arbitrary space dimensions. *Communications on Pure and Applied Mathematics*, 48(3) :235–276, 1995.

[62] Guillaume Latu, Michel Mehrenberger, Yaman Güçlü, Maurizio Ottaviani, and Eric Sonnendrücker. Field-aligned interpolation for semi-Lagrangian gyrokinetic simulations. *J. Sci. Comput.*, 74(3) :1601–1650, 2018.

[63] Peter Lax. Shock waves and entropy. In *Contributions to nonlinear functional analysis*, pages 603–634. Elsevier, 1971.

[64] Kévin Le Balc'h. Null-controllability of two species reaction-diffusion system with nonlinear coupling : a new duality method. *SIAM Journal on Control and Optimization*, 57(4) :2541–2573, 2019.

[65] Randall J. LeVeque. *Numerical methods for conservation laws*, volume 214. Springer, 1992.

[66] Xun Jing Li and Jiong Min Yong. Necessary conditions for optimal control of distributed parameter systems. *SIAM J. Control Optim.*, 29(4) :895–908, 1991.

[67] Eric Madaule, Sever Adrian Hirstoaga, Michel Mehrenberger, and Jérôme Pétri. Semi-Lagrangian simulations of the diocotron instability. Research report, July 2013.

[68] Idriss Mazari, Grégoire Nadin, and Ana Isis Toledo Marrero. Optimization of the total population size with respect to the initial condition in reaction-diffusion equations. Work in progress, 2021.

[69] Idriss Mazari, Domènec Ruiz-Balet, and Enrique Zuazua. Constrained control of bistable reaction-diffusion equations : gene-flow and spatially heterogeneous models. Preprint 2020.

[70] Robert W. McCormack. The effect of viscosity in hypervelocity impact cratering. *AIAA Paper*, (354), 1969.

[71] Robert I. McLachlan and G. Reinout W. Quispel. Splitting methods. *Acta Numerica*, 11 :341–434, 2002.

[72] Tiago Yuzo Miyaoka, Suzanne Lenhart, and João FCA Meyer. Optimal control of vaccination in a vector-borne reaction–diffusion model applied to Zika virus. *Journal of mathematical biology*, 79(3) :1077–1104, 2019.

[73] Michael S. Mock. Systems of conservation laws of mixed type. *Journal of Differential equations*, 37(1) :70–88, 1980.

[74] Grégoire Nadin and Ana Isis Toledo Marrero. On the maximization problem for solutions of reaction-diffusion equations with respect to their initial data. *Math. Model. Nat. Phenom.*, to appear, 2020.

[75] Hiroshi Otomo, Bruce M. Boghosian, and François Dubois. Two complementary lattice-Boltzmann-based analyses for nonlinear systems. *Physica A : Statistical Mechanics and its Applications*, 486 :1000–1011, 2017.

[76] Tiancheng Ouyang and Junping Shi. Exact multiplicity of positive solutions for a class of semilinear problem. II. *J. Differ. Equations*, 158(1) :94–151, 1999.

[77] Benoît Perthame. Boltzmann type schemes for gas dynamics and the entropy property. *SIAM Journal on Numerical Analysis*, 27(6) :1405–1421, 1990.

[78] Benoît Perthame. *Parabolic equations in biology. Growth, reaction, movement and diffusion.* Cham : Springer, 2015.

[79] Yue-Hong Qian, Dominique d'Humières, and Pierre Lallemand. Lattice BGK models for Navier-Stokes equation. *EPL (Europhysics Letters)*, 17(6) :479, 1992.

[80] Viktor V. Rusanov. The calculation of the interaction of non-stationary shock waves and obstacles. *USSR Computational Mathematics and Mathematical Physics*, 1(2) :304–320, 1962.

[81] J. Schraiber, A. Kaczmarczyk, R. Kwok, M. Park, R. Silverstein, F. Rutaganira, T. Aggarwal, M. Schwemmer, C. Hom, R. Grosberg, et al. Constraints on the use of lifespan-shortening *Wolbachia* to control dengue fever. *Journal of theoretical biology*, 297 :26–32, 2012.

[82] Magdi M. Shoucri. A two-level implicit scheme for the numerical solution of the linearized vorticity equation. *International Journal for Numerical Methods in Engineering*, 17(10) :1525–1538, 1981.

[83] Jacques Simon. Compact sets in the space $L^p(0,T;B)$. *Ann. Mat. Pura Appl. (4)*, 146 :65–96, 1987.

[84] Steven P. Sinkins. *Wolbachia* and cytoplasmic incompatibility in mosquitoes. *Insect biochemistry and molecular biology*, 34(7) :723–729, 2004.

[85] Eric Sonnendrücker, Jean Roche, Pierre Bertrand, and Alain Ghizzo. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *J. Comput. Phys.*, 149(2) :201–220, 1999.

[86] Benoit Stoll, Hervé Bossin, Hereiti Petit, Jerome Marie, and Michel Cheong Sang. Suppression of an isolated population of the mosquito vector Aedes polynesiensis on the atoll of Tetiaroa, French Polynesia, by sustained release of *Wolbachia*-incompatible male mosquitoes. In *Conference : ICE - XXV International Congress of Entomology, At Orlando, Florida, USA.*, 2016.

[87] A. Storelli, L. Vermare, P. Hennequin, Ö. D. Gürcan, G. Dif-Pradalier, Y. Sarazin, X. Garbet, T. Görler, R. Singh, P. Morel, V. Grandgirard, P. Ghendrih, and Tore Supra team. Comprehensive comparisons of geodesic acoustic mode characteristics and dynamics between tore supra experiments and gyrokinetic simulations. *Phys. Plasmas*, 22(6) :062508, 2015.

[88] Gilbert Strang. On the construction and comparison of difference schemes. *SIAM journal on numerical analysis*, 5(3) :506–517, 1968.

[89] Martin Strugarek and Nicolas Vauchelet. Reduction to a single closed equation for 2-by-2 reaction-diffusion systems of Lotka-Volterra type. *SIAM J. Appl. Math.*, 76(5) :2060–2080, 2016.

[90] Martin Strugarek, Nicolas Vauchelet, and Jorge Zubelli. Quantifying the survival uncertainty of *Wolbachia*-infected mosquitoes in a spatial model. *Math. Biosci. Eng.*, 15 :961–991, 2018.

[91] Sauro Succi. *The lattice Boltzmann equation : for fluid dynamics and beyond.* Oxford university press, 2001.

[92] Dean Thomas, Christl Donnelly, Roger Wood, and Luke Alphey. Insect population control using a dominant, repressible, lethal genetic system. *Science*, 287(5462) :2474–2476, 2000.

[93] Andreas Wächter and Lorenz T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.*, 106(1, Ser. A) :25–57, 2006.

[94] T. Walker, P.H. Johnson, L.A. Moreira, I. Iturbe-Ormaetxe, F.D. Frentiu, C.J. McMeniman, Y.S. Leong, Y. Dong, J. Axford, P. Kriesner, et al. The *w*Mel *Wolbachia* strain blocks dengue and invades caged *Aedes aegypti* populations. *Nature*, 476(7361) :450, 2011.

[95] Hans F. Weinberger. Invariant sets for weakly coupled parabolic and elliptic systems. *Rend. Mat*, 8(6) :295–310, 1975.

[96] John H. Werren, Laura Baldo, and Michael E. Clark. *Wolbachia* : master manipulators of invertebrate biology. *Nature Reviews Microbiology*, 6(10) :741, 2008.

[97] Raoyang Zhang, Hongli Fan, and Hudong Chen. A lattice Boltzmann approach for solving scalar transport equations. *Philos. Trans. R. Soc. Lond., A*, 369(1944) :2264–2273, 2011.

[98] Xiaoying Zheng, Dongjing Zhang, Yongjun Li, et al. Incompatible and sterile insect techniques combined eliminate mosquitoes. *Nature*, 572 :56–61, Aug 2019.

Nous nous intéressons à la résolution théorique et numérique d'un système de lois de conservation

$$\partial_t w + \nabla \cdot \boldsymbol{q}(w) = 0.$$

L'application envisagée concerne la simulation des plasmas de tokamak.
Pour résoudre numériquement cette équation, nous utilisons un schéma cinétique de relaxation $DdQn_v$, qui l'approche avec n_v équations et n_v inconnues cinétiques.
Les modèles cinétiques ont l'avantage d'être des schémas numériques efficaces basés sur la résolution d'équations de transport à vitesse constante. Cette thèse propose d'une part une nouvelle technique d'analyse de la stabilité de ces schémas. La méthode classique consiste à considérer une équation équivalente avec une inconnue w. Dans cette thèse, nous proposons un système équivalent avec n_v équations et n_v inconnues : l'inconnue w de notre loi de conservation et $n_v - 1$ variables supplémentaires.
L'hyperbolicité de ce système équivalent nous donne une condition pour obtenir une solution stable. Pour certains schémas cinétiques, cette condition peut différer de la condition sous-caractéristique diffusive déduite de l'équation équivalente classique. Nous nous intéressons également à la construction de conditions limites stables et avec une précision d'ordre 2.

Ces schémas de relaxation cinétique sont appliqués à un modèle de drift qui modélise le mouvement du plasma dans un tokamak. Nous proposons d'utiliser la structure des équations cinétiques pour construire un schéma de Galerkin Discontinu sans CFL, qui ne nécessite pas d'inversion de matrices. Nous appliquons cette méthode à des modélisations d'instabilités de Diocotron pour un plasma de tokamak.

INSTITUT DE RECHERCHE MATHÉMATIQUE AVANCÉE
UMR 7501
Université de Strasbourg et CNRS
7 Rue René Descartes
67 084 STRASBOURG CEDEX

Tél. 03 68 85 01 29
Fax 03 68 85 03 28

https://irma.math.unistra.fr
irma@math.unistra.fr

IRMA 2023/002
https://tel.archives-ouvertes.fr/tel-04034510

ISSN 0755-3390