



HAL
open science

Résolution des équations de Maxwell tridimensionnelles instationnaires sur architecture massivement multicoeur

Thomas Strub

► **To cite this version:**

Thomas Strub. Résolution des équations de Maxwell tridimensionnelles instationnaires sur architecture massivement multicoeur. Analyse numérique [math.NA]. Université de Strasbourg, 2015. Français. NNT : 2015STRAD050 . tel-01651258

HAL Id: tel-01651258

<https://theses.hal.science/tel-01651258>

Submitted on 28 Nov 2017

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.

*ÉCOLE DOCTORALE Mathématiques,
Sciences de l'Informatique et de l'ingénieur,*

Institut de Recherche Mathématiques Avancée, UMR 7501

THÈSE présentée par :

Thomas STRUB

soutenue le : 13 mars 2015

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

Discipline/ Spécialité : MATHÉMATIQUES APPLIQUÉES

**Résolution des équations de Maxwell
tridimensionnelles instationnaires sur
architecture massivement multicœur**

THÈSE dirigée par :

Mr. Philippe Helluy

Professeur, université de Strasbourg

RAPPORTEURS :

Mr. Stéphane Lanteri

Mr. Xavier Ferrières

Directeur de recherche, INRIA Sophia Antipolis

Maître de recherche HDR, ONERA Toulouse

EXAMINATEURS :

Mr. Christophe Guiffaut

Mr. Florian de Vuyst

Chargé de recherche CNRS, Xlim Limoges

Professeur, ENS Cachan

ENCADRANT INDUSTRIEL :

Mr. Christophe Girard

Axessim Illkirch

Remerciements

Mes remerciements vont d'abord à Philippe Helluy qui a encadré mon travail durant ces trois années. Sans ses grandes connaissances et sa disponibilité, cette thèse n'aurait jamais vu le jour. Travailler avec Philippe a été très enrichissant.

Je souhaite également remercier Stéphane Lanteri et Xavier Ferrières d'avoir accepté de rapporter ce manuscrit, ainsi que Christophe Guiffaut et Florian de Vuyst de faire partie du jury en tant qu'examinateur. Je remercie également les membres invités de mon jury, Eric Kerhervé, Christian Carel, Bernard Pecqueux et Christophe Prud'homme.

Je tiens également à remercier chaleureusement toute l'équipe d'Axessim qui m'a accueilli durant ces trois années. Je remercie Christophe Girard d'avoir encadré ma thèse et d'avoir partagé son expertise en électromagnétisme. Je remercie également tout le reste de l'équipe Didier, Cyril, Natahanael, Philippe, Guillaume, Xavier, Sophie, Charline et Lionel, pour leur disponibilité et l'aide précieuse qu'ils ont pu m'offrir.

Pour finir, je remercie ma famille qui m'a apporté son soutien durant ma thèse et également durant toutes mes études.

Table des matières

Introduction	3
1 Équations de Maxwell	7
1.1 Les équations de Maxwell	7
1.1.1 Formulation	8
1.1.2 Une variante : Maxwell avec correction de divergence .	10
1.2 Mise sous forme d'un système de Friedrichs	11
1.2.1 Équations de Maxwell	11
1.2.2 Équations de Maxwell avec correction de divergence .	14
1.3 Condition de saut	17
1.4 Une condition limite absorbante, les Perfectly Matched Layers	20
2 Formulation Galerkin Discontinue	27
2.1 Généralités sur les systèmes de Friedrichs	27
2.2 Formulation faible du problème d'évolution	30
2.3 Formulation Galerkin Discontinue	33
2.4 Stabilité et conditions aux limites	36
2.5 Formulation semi-discrète	40

2.6	Convergence du schéma semi-discret	41
3	Modèles physiques	49
3.1	Condition de conducteur électrique parfait	49
3.2	Matériau conducteur magnétique parfait	52
3.3	Condition aux limites avec impédance de surface (SIBC)	53
3.4	Condition aux limites de Silver-Müller	55
3.5	Modèle de Bérenger	56
3.6	Injection d'un champ incident	57
4	Interpolation Galerkin discontinue	61
4.1	Espace d'approximation	61
4.1.1	Maille de référence	62
4.1.2	Transformation géométrique	69
4.2	Approximation des intégrales	72
4.2.1	Terme de masse	73
4.2.2	Terme de rigidité	73
4.2.3	Terme de flux	75
4.2.4	Validité de l'intégration numérique	78
4.3	Schéma temporel Runge-Kutta 2	81
4.3.1	Stabilité du schéma temporel	81
4.3.2	Une condition de stabilité <i>a priori</i>	84
4.3.3	Une application du diagnostic numérique de stabilité	85
5	Programmation sur GPU	89

5.1	Parallélisation OpenCL	90
5.1.1	La bibliothèque OpenCL	90
5.1.2	Zone de maillage homogène	93
5.1.3	Implémentation générique	93
5.1.4	Optimisation sur maillage hexaédrique	99
5.2	Parallélisation MPI	105
6	Validations	109
6.1	Ordre de convergence de la méthode	109
6.2	Domaine de validité fréquentielle	113
6.3	Ligne microruban	117
6.4	Antenne cornet sectorale	121
6.5	Genec	124
6.5.1	Résultats	126
6.5.2	Performances	127
7	Algorithme Galerkin Discontinu parallèle asynchrone	133
7.1	Structure de zones	133
7.1.1	Zone homogène	134
7.1.2	Zone d'interface	135
7.1.3	Zone d'interface MPI	136
7.2	Déroulement des calculs	139
7.2.1	Un algorithme GD à deux niveaux	139
7.2.2	Graphe des tâches	139

7.2.3	Asynchronisation des transferts	140
8	Validations de l'algorithme asynchrone	147
8.1	Raffinement	147
8.1.1	Raffinement h	147
8.1.2	Raffinement hp	150
8.2	Scalabilité	151
8.2.1	Cas test	152
8.2.2	Résultats	152
9	Modèles de fils	157
9.1	Principe des modèles de fil	157
9.2	Prise en compte de la singularité	160
9.2.1	Modèle unidimensionnel	160
9.2.2	Source ponctuelle de courant	162
9.2.3	Application directe de la source	164
9.2.4	Approximation de Padé	164
9.2.5	Comparaison des résultats	166
10	Calcul de matrice de capacitance et inductance de section de fil	171
10.1	Problème physique	172
10.2	Schéma numérique	175
10.2.1	Équation intégrale	175
10.2.2	Discrétisation	178
10.3	Parallélisation	182

10.3.1	Calcul des matrices	182
10.3.2	Pivot de Gauss parallèle	182
10.4	Résultats	186
10.4.1	Convergence	187
10.4.2	Section complexe	187
	Conclusion	191

Introduction

La simulation numérique est de plus en plus utilisée dans différents domaines d'application. En particulier, la résolution numérique des équations de Maxwell instationnaires est employée dans par exemple la compatibilité électromagnétique. Actuellement, la méthode rencontrant le plus de succès est la méthode des différences finies (FDTD). Celle-ci est robuste, rapide et elle est compatible avec un grand nombre de modèles physiques. Toutefois, elle présente différents inconvénients. La discrétisation des structures en marches d'escalier peut entraîner une diminution de la précision de la solution obtenue. De plus, elle présente une forte dispersion, limitant la taille des scènes pouvant être modélisées. Depuis quelques années, des méthodes de type Galerkin Discontinue (GD) ont été développées. Ces méthodes permettent de représenter finement les structures simulées et offrent une bonne précision des calculs. Néanmoins, le coût de ces méthodes en temps et mémoire est important. Heureusement, la méthode GD se prête bien à la parallélisation.

Au cours de cette thèse, une méthode GD parallèle a été développée. Ces travaux ont été effectués dans le cadre du projet GREAT (Galerkin Resolution for Electromagnetic Applications in Time domain) financé par DGA/DS/MRIS dans le cadre d'un projet d'innovation duale RAPID. Nous nous basons sur des travaux effectués à l'ONERA et INRIA ayant conduit à la création d'un logiciel de recherche implémentant la méthode GD et appelé FemGD. Le projet GREAT a pour but la réécriture du logiciel et son optimisation, en vue d'une commercialisation future. La réécriture de la méthode passe par la mise en place d'un environnement de simulation reposant sur des outils de pré- et post-traitement adaptés, ainsi qu'à la validation de l'outil produit. Dans ce manuscrit, nous nous intéressons principalement à l'optimisation de la méthode sur un ou plusieurs GPU. Les cartes graphiques récentes ("Graphic Processing Units") sont en effet de plus en plus utilisées pour le calcul scientifique. Nous avons choisi OpenCL comme environnement de programmation pour les GPU et les processeurs multicœurs. Les communications entre GPU seront assurées par la bibliothèque MPI ("Message Passing Interface").

Le chapitre 1 de cette thèse est consacré à la présentation des équations de Maxwell ainsi qu'à l'introduction d'une correction de divergence. Cette correction de divergence ne sera pas utilisée dans la suite de cette thèse, mais elle permet d'écrire des équations un peu plus générales. Les systèmes d'équations aux dérivées partielles obtenus sont en effet sous la forme d'un système dit de Friedrichs. Nous rappelons différents résultats classiques sur ce type de système, notamment des conditions assurant l'existence et l'unicité de la solution portant sur les conditions limites. Nous établissons également des propriétés vérifiées par les discontinuités de la solution, mettant ainsi en lumière l'intérêt de l'approximation des champs par des fonctions discontinues. Enfin, nous décrivons un modèle de couches absorbantes permettant de simuler un domaine infini, les perfectly matched layers (PML), qui lui aussi admet une formulation de type système de Friedrichs. Nous mettons ainsi en évidence l'intérêt à disposer d'un solveur GD pour système de Friedrichs générique.

L'écriture sous forme générique permet de mettre en place une méthode de résolution pouvant s'appliquer à différents systèmes de lois de conservation. Dans le chapitre 2, nous établissons une formulation faible du système de Friedrichs faisant intervenir une solution et des fonctions tests discontinues, appelées formulation GD. Dans cette écriture apparaît un flux numérique sur les surfaces de discontinuité. Nous choisissons ici d'utiliser un flux décentré et discutons brièvement les raisons de ce choix. Nous présentons ensuite une discrétisation de cette formulation sur maillage hexaédrique déstructuré et donnons une condition de stabilité du schéma ainsi qu'une preuve de sa convergence. Nous étudions également la limite de validité dans le domaine fréquentiel de cette approximation en établissant une loi en « λ sur dix » similaire à celle connue en FDTD. Pour finir, nous définissons différents modèles physiques mis sous la forme de conditions limites vérifiant les conditions d'existence et d'unicité de la solution et de stabilité du schéma.

La méthode GD permet la prise en compte fine des structures simulées, mais au prix d'un coup de calcul important. Afin de pallier cet inconvénient, nous avons implémenté un algorithme parallèle de cette méthode en utilisant les bibliothèques OpenCL et MPI. Dans le chapitre 5, nous décrivons tout d'abord les spécificités de la parallélisation OpenCL sur processeur graphique (GPU). Puis nous décrivons une version générique et une version optimisée sur maillage hexaédrique de l'algorithme de calcul. L'utilisation de ce type de périphérique introduit une limitation de la taille des cas pouvant être traités, en effet la mémoire d'un GPU est de l'ordre de quelques gigaoctets. Nous parallélisons donc les calculs sur plusieurs GPU grâce à la bibliothèque MPI. Dans un premier temps, nous mettons en place une parallélisation synchrone des calculs. Celle-ci, ne se révélant pas optimale, nous conduira à asynchroniser les calculs et communications. Néanmoins, cette première

implémentation permet d'effectuer la simulation de structures complexes.

Différents cas tests ont été mis en place afin de valider le schéma GD parallèle résolvant les équations de Maxwell. Le chapitre 6 regroupe la description et les résultats obtenus dans cinq cas tests. Le premier cas test nous permet de valider l'algorithme en vérifiant son ordre de convergence et d'estimer le gain de temps dû à l'optimisation de l'algorithme. Dans le second cas test, nous cherchons à estimer numériquement la limite fréquentielle de validité de la méthode pour un maillage de pas fixé. Les trois cas tests suivants font partie des tests de validations utilisés dans le projet GREAT. Le premier de ces tests consiste en la simulation d'une ligne microruban et notamment le calcul de l'impédance de la ligne. Nous cherchons notamment à mettre en avant la précision de l'impédance calculée en comparant les résultats à ceux obtenus avec une méthode FDTD. Dans le second cas test, nous simulons une antenne cornet sectorale et calculons son diagramme de rayonnement ainsi que son rapport d'onde stationnaire (ROS). Enfin, le troisième cas test a pour but de quantifier l'efficacité de la parallélisation MPI synchrone. Nous simulons pour cela un missile générique (GENEC). Les résultats du calcul sont comparés avec ceux obtenus par la méthode FDTD et des résultats expérimentaux. Ce cas test montre également que la parallélisation MPI n'atteint pas les performances optimales.

Afin d'améliorer les performances et de mettre en place certaines fonctionnalités, nous définissons dans le chapitre 7 une seconde implémentation de la méthode GD parallèle. Cette implémentation s'appuie sur une répartition des mailles en zones homogènes sur lesquelles sont appliqués des *kernels* (programmes exécutés par le GPU) de calculs spécifiquement optimisés. De plus, cette structure de données permet de mettre en place un modèle de raccord non conforme entre zones en appliquant sur les interfaces des traitements spécifiques. De la même manière, cette structure permet d'asynchroniser les calculs et les transferts de données. Pour cela, nous définissons des tâches composant la suite de calculs et transferts présents dans l'avancée d'un pas de temps. Ces tâches forment un graphe orienté que nous décrivons au moyen d'événements OpenCL. Le lancement en parallèle de ces tâches permet alors l'asynchronisation des calculs et les liens de dépendances décrits par les événements OpenCL assurent la validité des calculs.

Dans le chapitre 8, nous validons cette seconde implémentation. Cette validation est faite au moyen de tests pour lesquels nous connaissons la solution exacte ce qui nous permet de valider simplement les calculs. D'abord nous effectuons un test portant sur la prise en compte des non-conformités. Nous calculons l'ordre de convergence de la méthode avec différents degrés d'approximation constants et non constants. Dans un second cas test, nous estimons l'accélération obtenue en parallélisant de manière asynchrone les calculs

sur plusieurs périphériques OpenCL. Nous comparons ce résultat avec l'accélération maximale et l'accélération de la parallélisation synchrone. Nous pouvons ainsi mettre en évidence le gain de performance obtenu et constatons que l'accélération est proche de la valeur maximale.

Les deux derniers chapitres sont consacrés à deux études, relativement indépendantes du reste de la thèse, sur des problématiques de couplages GD/fils conducteurs.

De nombreuses méthodes numériques pour coupler un solveur de champs électromagnétiques avec un modèle de fils conducteurs ont été développées. Ce type de couplages est fréquent en FDTD, voir par exemple [35] et [30]. En GD et volumes finis, différents travaux ont été menés pour définir des modèles similaires, voir [56, 72, 36]. Dans le chapitre 9, nous commençons par présenter le principe des méthodes de couplage des champs électromagnétiques avec une source de courant linéique, ainsi que la difficulté de représentation des champs rayonnés par une telle source. Au voisinage du fil, les champs sont en général singuliers, ce qui interdit une bonne précision de la méthode GD, même d'ordre élevé, à cause du phénomène d'oscillations de Gibbs. Nous présentons, dans un cas très simplifié, une technique pour éliminer la singularité due au fil. Quelques expériences numériques 1D viennent justifier l'approche.

Le dernier chapitre (chapitre 10) décrit une étude réalisée au début de la thèse. Il s'agit de calculer les matrices de capacitance et d'inductance d'une section de câbles. Ces matrices interviennent dans le modèle des lignes de transmissions qui permet de simuler la propagation du courant et de la charge dans les câbles. L'étude a consisté à accélérer le calcul des matrices sur GPU au moyen de la bibliothèque OpenCL. L'algorithme nécessite le calcul de l'inversion d'une matrice pleine. C'est principalement cette inversion, faite par l'algorithme du pivot de Gauss, que nous avons optimisée. Ces travaux ont permis d'optimiser un code de lignes de transmission écrit à l'ONERA.

Nous terminons ce manuscrit par une conclusion générale reprenant les différents résultats présentés. Nous donnons les ouvertures et perspectives dans la suite de ces travaux.

Chapitre 1

Équations de Maxwell

Dans ce premier chapitre, nous donnerons les formulations de deux systèmes de lois de conservation : les équations de Maxwell et les équations de Maxwell avec correction de divergence. Ces deux systèmes régissent la propagation des ondes électromagnétiques et permettent de modéliser un grand nombre de configurations. Le système avec correction de divergence introduit un couplage des équations de Maxwell avec un système annexe permettant de réduire les erreurs dues au non-respect des conditions de divergence des champs ((1.2a page suivante) et (1.2b page suivante)). Nous écrirons ensuite ces deux systèmes sous une forme commune, celle d'un système hyperbolique symétrique linéaire du premier ordre ou système de Friedrichs. De nombreux résultats classiques sont connus pour ce type de systèmes. Nous examinerons ensuite les discontinuités possibles des solutions de ces systèmes. Enfin, nous présenterons une technique permettant de résoudre un problème en domaine infini par l'ajout à la périphérie du domaine de calcul de couches absorbantes appelées Perfectly Matched Layers (PML). Nous verrons alors que dans les PML, le système obtenu peut également être écrit sous la forme d'un système de Friedrichs. Cela motivera la mise en place d'une méthode générale de résolution. Nous pourrions alors bien sûr résoudre les systèmes décrits dans ce chapitre, mais aussi tout système pouvant être écrit sous la forme d'un système de Friedrichs.

1.1 Les équations de Maxwell

Les équations de Maxwell sont l'œuvre du physicien écossais James Clerk Maxwell. Celles-ci décrivent les propagations des champs électromagnétiques dans un milieu quelconque. À la fin du XIXe siècle, Maxwell s'est basé sur

les travaux concernant l'électricité, le magnétisme et l'induction et les a unifiées en un système d'équations faisant intervenir les champs électrique et magnétique. Ces travaux furent poursuivis par Heaviside qui introduisit l'écriture sous forme d'équations aux dérivées partielles des équations de Maxwell. Dans un premier temps, nous présentons les équations de Maxwell en milieu linéaire. Nous donnerons ensuite une modification des équations de Maxwell introduisant une correction de la divergence des champs électrique et magnétique.

1.1.1 Formulation

Commençons par définir quelques notations qui nous seront utiles tout au long de ce manuscrit. Soient $\mathcal{E} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ et $\mathcal{H} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ les champs électrique et magnétique. Soit $\mathcal{J} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ le courant électrique. Notons les paramètres constitutifs des matériaux par ε , μ , σ et σ^* représentant respectivement la permittivité, la perméabilité et les conductivités électrique et magnétique du milieu.

Nous noterons également un point de l'espace $x = (x^1, x^2, x^3)$ et ∂_i la dérivée partielle suivant x^i . De plus, notons ∇ l'opérateur gradient.

Avec ces notations, en tenant compte des équations constitutives des milieux linéaires, les équations de Maxwell s'écrivent

$$\partial_t \varepsilon \mathcal{E} + \sigma \mathcal{E} - \nabla \times \mathcal{H} = -\mathcal{J}, \quad (1.1a)$$

$$\partial_t \mu \mathcal{H} + \sigma^* \mathcal{H} + \nabla \times \mathcal{E} = 0. \quad (1.1b)$$

L'équation (1.1a) est appelée équations de Maxwell-Ampère et l'équation (1.1b) est appelée équations de Maxwell-Faraday.

À ces deux équations s'ajoutent des conditions sur les divergences des champs électromagnétiques exprimées par les équations de Maxwell-Gauss (1.2a) et Maxwell-Thomson (1.2b) où $\rho : \mathbb{R}^3 \rightarrow \mathbb{R}$ est la densité de charge électrique :

$$\nabla \cdot \varepsilon \mathcal{E} = \rho, \quad (1.2a)$$

$$\nabla \cdot \mu \mathcal{H} = 0. \quad (1.2b)$$

Proposition 1. *Lorsque les charges et courants vérifient l'équation de conservation de la charge*

$$\partial_t \rho + \nabla \cdot (\mathcal{J} + \sigma \mathcal{E}) = 0, \quad (1.3)$$

si les équations de Maxwell-Gauss (1.2a page ci-contre) et Maxwell-Thomson (1.2b page précédente) sont vérifiées à l'instant initial, alors elles le sont à tous les instants.

Démonstration. Supposons que la charge et le courant soient liés par l'équation de conservation de la charge (1.3). Alors, en appliquant l'opérateur de divergence aux équations (1.1a page ci-contre) et (1.1b page précédente), nous obtenons

$$\begin{aligned} \partial_t \nabla \cdot \varepsilon \mathcal{E} &= -\sigma \mathcal{E} - \nabla \cdot \mathcal{J} = \partial_t \rho, \\ \partial_t \nabla \cdot \mu \mathcal{H} &= -\sigma^* \mathcal{H}. \end{aligned} \quad (1.4)$$

puisque la divergence d'un rotationnel est nulle.

En ajoutant aux équations différentielles précédentes les conditions initiales

$$\begin{aligned} \nabla \cdot \varepsilon \mathcal{E}(x, 0) &= \rho(x, 0), \\ \nabla \cdot \mu \mathcal{H}(x, 0) &= 0. \end{aligned} \quad (1.5)$$

nous obtenons le résultat recherché.

□

Ainsi, une solution des équations de Maxwell vérifiant les conditions de divergence (1.2 page ci-contre) à l'instant initial, les vérifie à chaque instant. Il n'est alors plus nécessaire de considérer explicitement ces relations. Toutefois, les conditions de divergence des champs électromagnétiques ne sont en général pas vérifiées par une solution approchée. En temps long, les erreurs de divergence ont tendance à s'amplifier. Ce type d'erreurs peut notamment survenir en présence de sources de courant subcellulaires, par exemple des particules ou des fils. Nous présentons maintenant une modification des équations de Maxwell, conçue pour diminuer sur le plan numérique les erreurs de divergence.

Remarque 1. *Nous ne résoudrons pas numériquement les équations de Maxwell avec correction de divergence dans cette thèse. Cela pourra faire l'objet*

de futurs travaux. Ce système est présenté dans le but de justifier la généralité du schéma que nous introduirons dans les chapitres suivants.

1.1.2 Une variante : Maxwell avec correction de divergence

Afin d'obtenir une solution numérique vérifiant les conditions de divergence (1.2 page 8), nous pouvons coupler les équations de Maxwell à un système annexe vérifié par deux champs scalaires λ_e et λ_h . Cette technique a été introduite dans [53] pour des problèmes de couplage entre les équations de Maxwell et Vlasov, et étudiée par différents auteurs [25, 12] et également étendue à la magnétohydrodynamique [41]. Nous considérerons ici le cas d'une correction de divergence purement hyperbolique des équations de Maxwell.

Avec la correction de divergence, le système (1.1 page 8) devient

$$\begin{aligned}
 \partial_t \varepsilon \mathcal{E} + \sigma \mathcal{E} - \nabla \times \mathcal{H} + \chi \nabla (\varepsilon \lambda_e) &= -\mathcal{J}, \\
 \partial_t \mu \mathcal{H} + \sigma^* \mathcal{H} + \nabla \times \mathcal{E} + \chi \nabla (\mu \lambda_h) &= 0, \\
 \partial_t \varepsilon \lambda_e + \chi \nabla \cdot (\varepsilon \mathcal{E}) &= \chi \rho, \\
 \partial_t \mu \lambda_h + \chi \nabla \cdot (\mu \mathcal{H}) &= 0.
 \end{aligned} \tag{1.6}$$

Le paramètre χ est une constante positive. Dans le cas où ce paramètre est nul, les deux systèmes ne sont plus couplés et nous retrouvons les équations de Maxwell classiques. Lorsque $\chi > 0$, le nouveau système est une extension du système de Maxwell. En effet, lorsque λ_e et λ_h sont nuls, le système permet de retrouver à la fois les équations de Maxwell-Ampère (1.1a), Maxwell-Faraday (1.1b), mais aussi (1.2a) et (1.2b). L'intérêt de ce nouveau système d'équations est que les erreurs de divergence des champs électromagnétiques sont transportées hors du domaine de calcul à une vitesse χ par les champs scalaires annexes λ_e et λ_h .

Les deux systèmes que nous avons présentés peuvent être mis sous une forme commune, celle d'un système de Friedrichs. Les deux prochaines sections seront consacrées à l'écriture sous cette forme. Nous utiliserons ensuite cette écriture pour établir la formulation Galerkin Discontinue (GD). L'intérêt de cette approche est de pouvoir adapter les travaux effectués sur la méthode GD à d'autres systèmes de loi de conservation en les écrivant sous la forme d'un système de Friedrichs.

1.2 Mise sous forme d'un système de Friedrichs

L'objectif de cette thèse est de développer une implémentation efficace d'un solveur Maxwell GD. Comme nous le verrons, ce développement est technique. Il est donc important de s'assurer de la possibilité future de généraliser ce travail à d'autres applications physiques, comme l'électromagnétisme non-linéaire ou certains modèles de plasmas. Nous allons donc commencer par écrire les équations de Maxwell, ou les équations de Maxwell avec correction de divergence, sous la forme mathématique assez générale d'un système hyperbolique. Les développements informatiques seront donc, avec des modifications mineures, applicables à tout système de cette forme. Tout d'abord, remarquons que l'on peut écrire les équations de Maxwell (1.1 page 8) et les équations de Maxwell associées à une correction de divergence (1.6 page ci-contre) sous la forme

$$\partial_t \mathcal{A}^t \mathcal{W} + \Sigma \mathcal{W} + \sum_{i=1}^3 \partial_i \mathcal{A}^i \mathcal{W} = \mathcal{S}. \quad (1.7)$$

Dans la suite, nous utiliserons la convention de sommation des indices répétés, *i.e.* nous noterons $\partial_i \mathcal{A}^i = \sum_{i=1}^3 \partial_i \mathcal{A}^i$.

Donnons alors la définition d'un système de Friedrichs introduite dans [26] et [18].

Définition 1. *Si la matrice $\mathcal{A} = \mathcal{A}^i n_i$ est symétrique pour toute direction n et si \mathcal{A}^t est symétrique définie positive, alors le système précédent est appelé système de Friedrichs.*

1.2.1 Équations de Maxwell

Dans le cas des équations de Maxwell (1.1a) (1.1b), le vecteur inconnu $\mathcal{W} = (\mathcal{E}, \mathcal{H})^T$ est le vecteur des champs électromagnétiques.

Les matrices \mathcal{A}^t , Σ , \mathcal{A}^1 , \mathcal{A}^2 , \mathcal{A}^3 se déduisent des équations de Maxwell et

valent

$$\mathcal{A}^t = \begin{pmatrix} \varepsilon & 0 & 0 & 0 & 0 & 0 \\ 0 & \varepsilon & 0 & 0 & 0 & 0 \\ 0 & 0 & \varepsilon & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{pmatrix}, \quad (1.8)$$

$$\Sigma = \begin{pmatrix} \sigma & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^* & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^* & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^* \end{pmatrix}, \quad (1.9)$$

$$\mathcal{A}^1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (1.10)$$

$$\mathcal{A}^2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (1.11)$$

$$\mathcal{A}^3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (1.12)$$

Nous constatons que les matrices \mathcal{A}^i sont symétriques et que \mathcal{A}^t est symétrique (et même diagonale) définie positive. Le système ainsi écrit est donc un système de Friedrichs. Il est classique de montrer, de manière générale,

qu'un tel système est hyperbolique, c'est-à-dire que $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ est diagonalisable avec des valeurs propres réelles, pour toute direction $n \in \mathbb{R}^3$ [28].

En effet, puisque \mathcal{A}^t est symétrique définie positive, il existe une matrice D_0 diagonale réelle et une matrice P_0 unitaire ($P_0^{-1} = P_0^T$) telles que $\mathcal{A}^t = P_0^T D_0 P_0$. Nous pouvons alors définir

$$\sqrt{\mathcal{A}^t} = P_0^T \sqrt{D_0} P_0. \quad (1.13)$$

De plus, puisque $\mathcal{A}^i n_i$ est symétrique, il existe une D diagonale réelle et une matrice P unitaire telles que $\mathcal{A}^i n_i = P^T D P$. Alors,

$$\begin{aligned} \sqrt{\mathcal{A}^t} \mathcal{A}^{t-1} \mathcal{A}^i n_i \sqrt{\mathcal{A}^t} &= \sqrt{\mathcal{A}^t}^{-1} \mathcal{A}^i n_i \sqrt{\mathcal{A}^t} \\ &= (\sqrt{\mathcal{A}^t} P)^{-1} D (\sqrt{\mathcal{A}^t} P). \end{aligned} \quad (1.14)$$

Donc $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ est bien diagonalisable avec des valeurs propres réelles.

Dans notre cas, il est aussi possible de mener directement le calcul.

Proposition 2. *Soit c la vitesse de la lumière*

$$c = (\varepsilon\mu)^{-\frac{1}{2}}. \quad (1.15)$$

La matrice $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ est diagonalisable pour tout vecteur n et ses valeurs propres sont 0 , $-|n|c$ et $|n|c$ chacune de multiplicité deux.

Démonstration. Le polynôme caractéristique de $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ vaut

$$P(X) = X^2 \left(X^2 - \frac{|n|^2}{\varepsilon\mu} \right). \quad (1.16)$$

Et donc, puisque la célérité de la lumière vaut $c = (\varepsilon\mu)^{-\frac{1}{2}}$, les valeurs propres de $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ sont 0 , $-|n|c$ et $|n|c$ chacune de multiplicité deux.

Les vecteurs propres associés à la valeur propre 0 sont

$$\begin{aligned} &(n_1, n_2, n_3, 0, 0, 0) \text{ et} \\ &(0, 0, 0, n_1, n_2, n_3). \end{aligned} \quad (1.17)$$

Les vecteurs propres associés à la valeur propre $-|n|c$ sont

$$\begin{aligned} & (cn_3, 0, -cn_1, -\frac{n_1n_2}{|n|\varepsilon}, \frac{n_1^2 + n_3^2}{|n|\varepsilon}, -\frac{n_3n_2}{|n|\varepsilon}) \text{ et} \\ & (0, cn_3, -cn_2, -\frac{n_2^2 + n_3^2}{|n|\varepsilon}, \frac{n_1n_2}{|n|\varepsilon}, \frac{n_1n_3}{|n|\varepsilon}). \end{aligned} \quad (1.18)$$

Les vecteurs propres associés à la valeur propre $|n|c$ sont

$$\begin{aligned} & (cn_3, 0, -cn_1, \frac{n_1n_2}{|n|\varepsilon}, -\frac{n_1^2 + n_3^2}{|n|\varepsilon}, \frac{n_2n_3}{|n|\varepsilon}) \text{ et} \\ & (0, cn_3, -cn_2, \frac{n_2^2 + n_3^2}{|n|\varepsilon}, -\frac{n_1n_2}{|n|\varepsilon}, -\frac{n_1n_3}{|n|\varepsilon}). \end{aligned} \quad (1.19)$$

Ces vecteurs forment une base de \mathbb{R}^6 , ainsi la matrice $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ est diagonalisable.

□

1.2.2 Équations de Maxwell avec correction de divergence

Dans le cas des équations de Maxwell avec correction de divergence (1.6), le vecteur des inconnues est le vecteur $\mathcal{W} = (\mathcal{E}, \mathcal{H}, \lambda_h, \lambda_e)^T$. Dans ce cas, les matrices $\mathcal{A}^t, \Sigma, \mathcal{A}^1, \mathcal{A}^2, \mathcal{A}^3$ s'écrivent

$$\mathcal{A}^t = \begin{pmatrix} \varepsilon & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \varepsilon & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \varepsilon & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \varepsilon & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu \end{pmatrix}, \quad (1.20)$$

$$\Sigma = \begin{pmatrix} \sigma & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma^* & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma^* & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma^* & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (1.21)$$

$$\mathcal{A}^1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & \chi\varepsilon & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \chi\mu \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \chi\varepsilon & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \chi\mu & 0 & 0 & 0 & 0 \end{pmatrix}, \quad (1.22)$$

$$\mathcal{A}^2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \chi\varepsilon & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \chi\mu \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \chi\varepsilon & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \chi\mu & 0 & 0 & 0 \end{pmatrix}, \quad (1.23)$$

$$\mathcal{A}^3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \chi\varepsilon & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \chi\mu \\ 0 & 0 & \chi\varepsilon & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \chi\mu & 0 & 0 \end{pmatrix}. \quad (1.24)$$

C'est bien un système de Friedrichs. Il est donc hyperbolique, selon la théorie générale. Le calcul des valeurs propres et des vecteurs propres peut être mené analytiquement :

Proposition 3. *La matrice $\mathcal{A}^{t-1}\mathcal{A}^i n_i$ est diagonalisable pour tout vecteur n et ses valeurs propres sont $-|n|c$, $|n|c$, $-|n|\chi$, $|n|\chi$ chacune de multiplicité 2. Les vecteurs propres associés à la valeur propre $-|n|c$ sont*

$$\left(n_3, 0, -n_1, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_1 n_2}{|n|}, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_1^2 + n_3^2}{|n|}, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_2 n_3}{|n|}, 0, 0 \right) \quad (1.25)$$

et

$$\left(0, n_3, -n_2, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_2^2 + n_3^2}{|n|}, \sqrt{\frac{\varepsilon}{\mu}} \frac{n_1 n_2}{|n|}, \sqrt{\frac{\varepsilon}{\mu}} \frac{n_1 n_3}{|n|}, 0, 0 \right). \quad (1.26)$$

Les vecteurs propres associés à la valeur propre $|n|c$ sont

$$\left(n_3, 0, -n_1, \sqrt{\frac{\varepsilon}{\mu}} \frac{n_1 n_2}{|n|}, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_1^2 + n_3^2}{|n|}, \sqrt{\frac{\varepsilon}{\mu}} \frac{n_2 n_3}{|n|}, 0, 0 \right) \quad (1.27)$$

et

$$\left(0, n_3, -n_2, \sqrt{\frac{\varepsilon}{\mu}} \frac{n_2^2 + n_3^2}{|n|}, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_1 n_2}{|n|}, -\sqrt{\frac{\varepsilon}{\mu}} \frac{n_1 n_3}{|n|}, 0, 0 \right). \quad (1.28)$$

Les vecteurs propres associés à la valeur propre $-|n|\chi$ sont

$$(n_1, n_2, n_3, 0, 0, 0, -|n|, 0) \quad (1.29)$$

et

$$(0, 0, 0, n_1, n_2, n_3, 0, -|n|). \quad (1.30)$$

Les vecteurs propres associés à la valeur propre $|n|\chi$ sont

$$(n_1, n_2, n_3, 0, 0, 0, |n|, 0) \quad (1.31)$$

et

$$(0, 0, 0, n_1, n_2, n_3, 0, |n|). \quad (1.32)$$

Nous constatons que dans ce système modifié, la valeur propre nulle a disparu. C'est cette propriété qui assure que les erreurs de divergence seront transportées vers le bord au lieu de « stagner » dans le domaine de calcul.

1.3 Condition de saut

Nous allons maintenant examiner les possibles discontinuités apparaissant dans les solutions des deux systèmes de lois de conservation présentés précédemment. Les discontinuités apparaissant dans une solution d'un système hyperbolique vérifient les conditions de saut de Rankine-Hugoniot [65]. Appliquons ces conditions aux équations de Maxwell. Pour cela, considérons une surface spatio-temporelle $D \subset \mathbb{R}^4$ régulière orientée par une normale notée $N = (n_t, n)^T$. Le vecteur $n \in \mathbb{R}^3$ est la partie spatiale de la normale et $n_t \in \mathbb{R}$ est la partie temporelle de la normale. Si la partie spatiale n est un vecteur unitaire, c'est à dire si $|n| = 1$, alors $n_t = -u$ où u est la vitesse de la discontinuité dans sa direction normale.

La discontinuité sépare \mathbb{R}^4 en deux ouverts Ω_L et Ω_R . Nous supposons que n est orientée de L vers R . Soit alors \mathcal{W} solution des équations de Maxwell, présentant une discontinuité sur la surface D . Dans les deux ouverts Ω_L et Ω_R , \mathcal{W} est solution \mathcal{C}^1 des équations de Maxwell. Elle vérifie la condition de Rankine-Hugoniot sur la discontinuité. Notons \mathcal{W}_L , respectivement \mathcal{W}_R , les champs électromagnétiques restreints à Ω_L , respectivement Ω_R . De la même manière les paramètres physiques des milieux entourant D sont notés avec un indice L ou R .

Proposition 4. *La relation de Rankine-Hugoniot est satisfaite sur la discontinuité si et seulement si*

$$n_t(\mathcal{A}_L^t \mathcal{W}_L - \mathcal{A}_R^t \mathcal{W}_R) + ((\mathcal{A}^i n_i)_L \mathcal{W}_L - (\mathcal{A}^i n_i)_R \mathcal{W}_R) = 0. \quad (1.33)$$

En appliquant cette relation aux équations de Maxwell, nous déduisons la propriété suivante.

Proposition 5. *De part et d'autre de la discontinuité, les champs vérifient*

$$\begin{aligned} u(\varepsilon_L \mathcal{E}_L - \varepsilon_R \mathcal{E}_R) &= n \times (\mathcal{H}_R - \mathcal{H}_L), \\ u(\mu_L \mathcal{H}_L - \mu_R \mathcal{H}_R) &= n \times (\mathcal{E}_L - \mathcal{E}_R), \end{aligned} \quad (1.34)$$

où u est la vitesse de la discontinuité.

Pour mieux situer cette condition du point de vue physique, supposons dans un premier temps que le milieu diélectrique est homogène, *i.e.* $\varepsilon_L = \varepsilon_R = \varepsilon$ et $\mu_L = \mu_R = \mu$. Alors,

$$u \begin{pmatrix} \mathcal{E}_L - \mathcal{E}_R \\ \mathcal{H}_L - \mathcal{H}_R \end{pmatrix} = \begin{pmatrix} 0 & -\frac{1}{\varepsilon} n \times \\ \frac{1}{\mu} n \times & 0 \end{pmatrix} \begin{pmatrix} \mathcal{E}_L - \mathcal{E}_R \\ \mathcal{H}_L - \mathcal{H}_R \end{pmatrix}. \quad (1.35)$$

Le vecteur $(\mathcal{E}_L - \mathcal{E}_R, \mathcal{H}_L - \mathcal{H}_R)^T$ est donc vecteur propre associé à la valeur propre u de la matrice apparaissant dans l'équation précédente.

Les valeurs propres de cette matrice sont $-c = -(\varepsilon\mu)^{-\frac{1}{2}}$, 0 et $c = (\varepsilon\mu)^{-\frac{1}{2}}$, chacune de multiplicité deux.

Dans le cas où $u = \pm c$, la discontinuité se propage à la vitesse de la lumière. Ce type de discontinuité apparaît dans le cas où la condition initiale est discontinue ou pour des sources électromagnétiques de type mesure de Dirac. Il est alors bien connu que les méthodes d'ordre élevé ont du mal à reproduire correctement ces solutions (oscillations de Gibbs). La méthode GD s'adresse plutôt à des cas où les solutions ne présentent pas ce type de singularités. En pratique, ces deux cas de figure ne se présenteront pas, puisque physiquement ce type de discontinuité n'apparaît pas.

Dans le cas où la discontinuité est fixe, l'égalité obtenue se réduit à une condition de continuité des champs tangents à D . Il n'est donc pas possible d'observer des solutions discontinues sauf si la condition initiale est elle-même discontinue. À l'instant initial, les conditions de divergence (1.2a) et (1.2b) réduisent les possibilités d'observer effectivement des solutions discontinues. Si la charge ρ est assez régulière par exemple, cela impose que les sauts des composantes normales de \mathcal{E} et \mathcal{H} sont nuls à l'instant initial. À tous les temps, les champs ne peuvent donc pas présenter de sauts tangentiels ou normaux. Par conséquent, ils ne peuvent qu'être continus.

Dans le cas où les paramètres du milieu sont discontinus, la situation est différente. À l'instant initial, nous avons

$$\begin{aligned} \varepsilon_R \mathcal{E}_R \cdot n - \varepsilon_L \mathcal{E}_L \cdot n &= 0, \\ \mu_R \mathcal{H}_R \cdot n - \mu_L \mathcal{H}_L \cdot n &= 0. \end{aligned}$$

Par conséquent, la condition initiale est forcément discontinue au niveau des changements de matériaux ε et μ .

Nous arrivons à la conclusion que pour des matériaux inhomogènes, sur une discontinuité fixe, les champs électrique et magnétique tangents à la discon-

tinuité sont continus et leurs composantes normales présentent forcément une discontinuité. Ce type de discontinuité apparait aux interfaces entre matériaux présentant des paramètres physiques différents. Un avantage de la méthode GD comparée à d'autres méthodes est de pouvoir prendre en compte naturellement ces discontinuités fixes.

Intéressons-nous maintenant, à la relation de Rankine-Hugoniot appliquée aux équations de Maxwell avec correction de divergence. Comme dans le cas des équations de Maxwell sans correction de divergence, nous pouvons étudier les possibles discontinuités de la solution en appliquant la condition de Rankine-Hugoniot. Dans ce cas, cela conduit à la propriété suivante.

Proposition 6. *De part et d'autre des discontinuités, les champs vérifient*

$$\begin{aligned}
u(\varepsilon_L \mathcal{E}_L - \varepsilon_R \mathcal{E}_R) &= n \times (\mathcal{H}_R - \mathcal{H}_L) + \chi n (\varepsilon_L \lambda_{eL} - \varepsilon_R \lambda_{eR}), \\
u(\mu_L \mathcal{H}_L - \mu_R \mathcal{H}_R) &= n \times (\mathcal{E}_L - \mathcal{E}_R) + \chi n (\mu_L \lambda_{hL} - \mu_R \lambda_{hR}), \\
u(\varepsilon_L \lambda_{eL} - \varepsilon_R \lambda_{eR}) &= \chi n \cdot (\varepsilon_R \mathcal{E}_R - \varepsilon_L \mathcal{E}_L), \\
u(\mu_L \lambda_{hL} - \mu_R \lambda_{hR}) &= \chi n \cdot (\mu_R \mathcal{H}_R - \mu_L \mathcal{H}_L)
\end{aligned} \tag{1.36}$$

où u est la vitesse de la discontinuité.

En menant la même analyse que pour le cas Maxwell, dans un milieu homogène les discontinuités se propagent soit à la vitesse de la lumière c soit à la vitesse χ . Il n'y a plus de possibilité de discontinuité fixe, car le système n'a plus de valeur propre nulle.

En présence d'une discontinuité fixe de diélectriques, les champs vérifient

$$\begin{aligned}
n \times (\mathcal{H}_R - \mathcal{H}_L) + \chi n (\varepsilon_L \lambda_{eL} - \varepsilon_R \lambda_{eR}) &= 0, \\
n \times (\mathcal{E}_L - \mathcal{E}_R) + \chi n (\mu_L \lambda_{hL} - \mu_R \lambda_{hR}) &= 0, \\
\chi n \cdot (\varepsilon_R \mathcal{E}_R - \varepsilon_L \mathcal{E}_L) &= 0, \\
\chi n \cdot (\mu_R \mathcal{H}_R - \mu_L \mathcal{H}_L) &= 0.
\end{aligned} \tag{1.37}$$

Lorsque $\lambda_e = \lambda_h = 0$ nous retrouvons les mêmes conditions de saut que pour le système de Maxwell, avec prise en compte de la condition initiale. L'avantage de cette formulation est de maintenir une bonne précision des conditions de divergence des champs, même pour des temps longs et pour des termes sources assez raides.

Nous venons de mettre en évidence que les équations de Maxwell constituent un cas particulier de système de lois de conservation. Nous avons aussi présenté une variante du système de Maxwell, le modèle de Maxwell avec correction de divergence, qui présente des caractéristiques intéressantes pour

les applications numériques. Il sera donc important dans la suite de cette thèse de proposer des méthodes numériques ayant une certaine généralité pour pouvoir prendre en compte divers types de systèmes d'EDP.

Remarque 2. *Nous n'avons pas programmé la résolution des équations de Maxwell avec correction de divergence, cela pourra faire l'objet de futurs travaux.*

Nous terminons ce chapitre d'introduction des modèles par la présentation d'une technique de conditions aux limites transparentes aux frontières du domaine de calcul. En effet, dans de nombreuses applications, le domaine de simulation est infini. Il est donc indispensable d'appliquer des conditions artificielles sur un domaine de calcul borné. Dans la mesure du possible, ces conditions doivent assurer que la solution calculée soit proche de la solution en domaine infini. On parle alors de conditions transparentes. En 1994, Bérenger [3] a publié une méthode particulièrement efficace pour assurer la transparence de la condition aux limites. Cette méthode est basée sur la construction de couches de matériaux fictifs qui absorbent les ondes sans les réfléchir. Ces matériaux ont été baptisés Perfectly Matched Layers (PML). La méthode a ensuite été étendue et analysée théoriquement dans une vaste littérature (voir par exemple [49, 51, 52, 48, 46, 68, 71, 27, 23].)¹ Nous allons voir que le modèle PML est lui aussi une extension du modèle de Maxwell usuel, dans lequel le nombre d'inconnues est multiplié par deux. Il en résultera un nouveau système hyperbolique. C'est un argument supplémentaire en faveur de notre stratégie de développement de méthodes numériques générales et efficaces pour approcher les systèmes hyperboliques, et pas seulement le système de Maxwell standard.

1.4 Une condition limite absorbante, les Perfectly Matched Layers

Nous présentons brièvement le principe de construction du modèle PML. Il est plus simple d'établir le modèle après une transformée de Fourier-Laplace en temps. Nous considérons donc les équations de Maxwell harmoniques (1.38 page suivante) dans un pavé droit $\Omega = [x_{\min}^1, x_{\max}^1] \times [x_{\min}^2, x_{\max}^2] \times [x_{\min}^3, x_{\max}^3]$ inclus dans \mathbb{R}^3 . Il est possible d'établir des modèles PML pour des domaines de formes plus complexes. Voir par exemple [49]. Nous allons

1. Les PML ont eu beaucoup de succès : le papier original de Bérenger est cité plus de 7000 fois dans Google Scholar. À titre de comparaison, l'article original sur la FFT de Cooley et Tukey est cité environ 11000 fois...

entourer Ω de couches PML. Dans Ω , après transformée de Laplace en temps, les équations de Maxwell deviennent

$$\begin{aligned} p\varepsilon\mathcal{E} - \nabla \times \mathcal{H} &= 0, \\ p\mu\mathcal{H} + \nabla \times \mathcal{E} &= 0. \end{aligned} \tag{1.38}$$

où $p = i\omega + \alpha$ est un nombre complexe avec $\alpha \geq 0$ et ω la pulsation.

À l'extérieur de Ω , les couches PML correspondent formellement à un changement de variables complexes. Plus précisément, nous allons introduire de nouvelles variables $\hat{x} = (\hat{x}^1, \hat{x}^2, \hat{x}^3) \in \mathbb{C}^3$ et allons écrire les équations de Maxwell dans ces nouvelles variables. Mathématiquement, cette manipulation peut sembler assez osée ! Néanmoins, il est possible de la justifier théoriquement grâce à des outils avancés de géométrie différentielle sur des variétés complexes. Voir par exemple [51], [49] ou [48].

Le changement de variables s'écrit

$$\begin{aligned} \hat{x}^1 &= x^1 + \frac{1}{p} \int_0^{x^1} \sigma_1(s) ds, \\ \hat{x}^2 &= x^2 + \frac{1}{p} \int_0^{x^2} \sigma_2(s) ds, \\ \hat{x}^3 &= x^3 + \frac{1}{p} \int_0^{x^3} \sigma_3(s) ds. \end{aligned} \tag{1.39}$$

Les paramètres $\sigma_1, \sigma_2, \sigma_3$ sont homogènes à des conductivités, nous les appellerons conductivités fictives. On suppose que ces conductivités sont nulles dans Ω . Nous donnerons en (1.53 page 25) un choix possible pour ces différents paramètres. Notons τ le changement de variable décrit précédemment. Remarquons que sa matrice jacobienne τ' est diagonale.

Nous remplaçons le système de Maxwell original (1.38) par le système de Maxwell avec les dérivées formelles complexes.

$$p\varepsilon\hat{\mathcal{E}} - \hat{\nabla} \times \hat{\mathcal{H}} = 0, \tag{1.40a}$$

$$p\mu\hat{\mathcal{H}} + \hat{\nabla} \times \hat{\mathcal{E}} = 0. \tag{1.40b}$$

Il est important de souligner que nous ne partons pas des équations de Maxwell en domaine infini, mais bel et bien des équations de Maxwell modifiées par les couches PML. L'idée fondamentale est de passer de façon régulière du milieu sans absorption au milieu PML, ce qui implique une progression régulière des paramètres d'absorption σ dans les PML. Un argument de type développement de Taylor, ou développement analytique, permet alors de conclure que jusqu'à un certain ordre, la solution calculée est bien la restriction à un domaine borné de la solution en milieu libre.

En exprimant les opérateurs de dérivations (formels) à coefficients complexes $\hat{\partial}_1, \hat{\partial}_2, \hat{\partial}_3$ au moyen des opérateurs de dérivation $\partial_1, \partial_2, \partial_3$, nous constatons que ce nouveau système coïncide avec le système original sur Ω .

Posons

$$\begin{aligned} s_1 &= x^1 + \frac{1}{p}\sigma_1(x^1), \\ s_2 &= x^2 + \frac{1}{p}\sigma_2(x^2), \\ s_3 &= x^3 + \frac{1}{p}\sigma_3(x^3). \end{aligned} \tag{1.41}$$

Nous effectuons le changement de variables formel $\hat{\partial}_1 = \frac{1}{s_1}\partial_1$, $\hat{\partial}_2 = \frac{1}{s_2}\partial_2$ et $\hat{\partial}_3 = \frac{1}{s_3}\partial_3$.

Ainsi, nous obtenons l'expression du rotationnel d'un champ E

$$\hat{\nabla} \times E = \text{jac}(\tau^{-1})\tau'^T \nabla \times (\tau'^T E). \tag{1.42}$$

En injectant ces formules dans le système de Maxwell complexe, nous obtenons le système perturbé

$$\begin{aligned} p\varepsilon \text{jac}(\tau)\tau'^{-1T}\tau'^{-1T}\tilde{\mathcal{E}} - \nabla \times \tilde{\mathcal{H}} &= 0 \\ p\mu \text{jac}(\tau)\tau'^{-1T}\tau'^{-1T}\tilde{\mathcal{H}} + \nabla \times \tilde{\mathcal{E}} &= 0 \end{aligned} \tag{1.43}$$

où $\tilde{\mathcal{E}} = \tau'^T \mathcal{E}$ et $\tilde{\mathcal{H}} = \tau'^T \mathcal{H}$.

Dans Ω , le système (1.43) coïncide avec le système original.

La matrice $\text{jac}(\tau)\tau'^{-1T}\tau'^{-1T}$ est une matrice diagonale s'exprimant par

$$\text{jac}(\tau)\tau'^{-1T}\tau'^{-1T} = \text{diag}\left(\frac{s_2s_3}{s_1}, \frac{s_1s_3}{s_2}, \frac{s_1s_2}{s_3}\right). \quad (1.44)$$

Définissons alors les matrices S_1 , S_2 et S_3 par

$$\begin{aligned} S_1 &= \text{diag}(\sigma_1, \sigma_2, \sigma_3, \sigma_1, \sigma_2, \sigma_3), \\ S_2 &= \text{diag}(\sigma_2, \sigma_3, \sigma_1, \sigma_2, \sigma_3, \sigma_1), \\ S_3 &= \text{diag}(\sigma_3, \sigma_1, \sigma_2, \sigma_3, \sigma_1, \sigma_2). \end{aligned} \quad (1.45)$$

et les matrices N_1 , N_2 et N_3 par

$$\begin{aligned} N_1 &= \text{diag}(s_1, s_2, s_3, s_1, s_2, s_3) = I + \frac{1}{p}S_1, \\ N_2 &= \text{diag}(s_2, s_3, s_1, s_2, s_3, s_1) = I + \frac{1}{p}S_2, \\ N_3 &= \text{diag}(s_3, s_1, s_2, s_3, s_1, s_2) = I + \frac{1}{p}S_3. \end{aligned} \quad (1.46)$$

Nous pouvons aussi écrire le système (1.43 page précédente) sous la forme

$$p\mathcal{A}^t N_1^{-1} N_2 N_3 W + \mathcal{A}^i \partial_i W = 0, \quad (1.47)$$

\mathcal{A}^t étant la matrice diagonale définie par $\mathcal{A}^t = \text{diag}(\varepsilon_0, \varepsilon_0, \varepsilon_0, \mu_0, \mu_0, \mu_0)$.

Il s'agit maintenant, par une transformation de Laplace inverse de relever alors en temps ce système. Pour cela nous décomposons la matrice $pN_1^{-1}N_2N_3$ en élément simple.

$$\begin{aligned} pN_1^{-1}N_2N_3 &= (pI + S_1)^{-1} (pI + S_2) (pI + S_3) \\ &= pI + A + R_1 (pI + C)^{-1} R_2. \end{aligned} \quad (1.48)$$

Les matrices apparaissant dans cette formule sont données par

$$\begin{aligned}
A &= \text{diag}(\sigma_2 + \sigma_2 - \sigma_1, \sigma_1 + \sigma_3 - \sigma_2, \sigma_1 + \sigma_2 - \sigma_3, \\
&\quad \sigma_3 + \sigma_2 - \sigma_1, \sigma_1 + \sigma_3 - \sigma_2, \sigma_1 + \sigma_2 - \sigma_3), \\
C &= \text{diag}(\sigma_1, \sigma_2, \sigma_3, \sigma_1, \sigma_2, \sigma_3), \\
R_1 &= \text{diag}(\sigma_1 - \sigma_2, \sigma_3 - \sigma_2, \sigma_2 - \sigma_3, \sigma_1 - \sigma_2, \sigma_3 - \sigma_2, \sigma_2 - \sigma_3), \\
R_2 &= \text{diag}(\sigma_1 - \sigma_3, \sigma_1 - \sigma_2, \sigma_1 - \sigma_3, \sigma_1 - \sigma_3, \sigma_1 - \sigma_2, \sigma_1 - \sigma_3).
\end{aligned} \tag{1.49}$$

Si nous appliquons une transformée de Laplace inverse au système (1.48) nous tombons sur un système intégro-différentiel en temps. Nous introduisons donc une variable intermédiaire \mathcal{V} pour nous ramener à un système d'EDP en espace-temps.

Le système relevé en temps devient

$$\begin{aligned}
\mathcal{A}^t \partial_t \mathcal{W} + A \mathcal{A}^t \mathcal{W} + R_1 \mathcal{A}^t \mathcal{V} + \partial_i \mathcal{A}^i \mathcal{W} &= 0, \\
\partial_t \mathcal{V} + C \mathcal{V} - R_2 \mathcal{W} &= 0.
\end{aligned} \tag{1.50}$$

Le système (1.50) est un système de Friedrichs. En effet, posons

$$\begin{aligned}
\overline{\mathcal{W}} &= (\mathcal{W}, \mathcal{V})^T, \\
\overline{\mathcal{A}}^t &= \begin{pmatrix} g' \mathcal{A}^t & 0 \\ 0 & g' I \end{pmatrix}, \\
\overline{R} &= \begin{pmatrix} A \mathcal{A}^t & R_1 \mathcal{A}^t \\ -R_2 & C \end{pmatrix}, \\
\overline{\mathcal{A}}^i \partial_i &= \begin{pmatrix} \mathcal{A}^i \partial_i & 0 \\ 0 & 0 \end{pmatrix}.
\end{aligned} \tag{1.51}$$

Alors, le système (1.50) s'écrit sous la forme

$$\overline{\mathcal{A}}^t \partial_t \overline{\mathcal{W}} + \overline{R} \overline{\mathcal{W}} + \partial_i \overline{\mathcal{A}}^i \overline{\mathcal{W}} = 0. \tag{1.52}$$

Les conductivités sont données par la formule suivante faisant intervenir la profondeur des PML δ_{\max} et un paramètre entier m définissant la croissance

des conductivités

$$\sigma_i(x^i) = \sigma_{\max} \left(\frac{\delta_i(x^i)}{\delta_{\max}} \right)^m \quad (1.53)$$

où $\delta_i(x^i)$ est la distance de x^i au domaine Maxwell soit

$$\delta_i(x^i) = \begin{cases} \min(x_{\min}^i - x^i, x^i - x_{\max}^i) & , \text{ si } x^i \leq x_{\min}^i \text{ ou } x^i \geq x_{\max}^i, \\ 0 & , \text{ sinon.} \end{cases} \quad (1.54)$$

Ce choix implique une certaine régularité de l'absorption PML. Comme déjà expliquée plus haut, cette régularité est importante pour assurer la « transparence » des PML.

Conclusion

En conclusion, dans ce chapitre d'introduction, nous avons d'abord rappelé les équations de Maxwell. Nous avons montré qu'elles constituent un système hyperbolique de Friedrichs. En toute rigueur, le modèle de Maxwell n'est pas constitué que d'un système hyperbolique sur le champ électrique et le champ magnétique. Il contient aussi une condition initiale sur la divergence de ces champs.

Nous avons alors introduit une extension du modèle de Maxwell, le modèle de Maxwell avec correction de divergence de [53]. C'est encore un système hyperbolique. Il est plus facile à approcher numériquement, car il permet de maintenir au cours du temps les conditions sur les divergences des champs. Ce modèle est également compatible avec les relations de sauts physiques du système de Maxwell classique.

Nous avons enfin rappelé comment construire une condition aux limites transparentes au moyen des couches PML de Bérenger. À l'intérieur des couches absorbantes, le modèle électromagnétique obtenu se présente encore sous la forme d'un système hyperbolique.

Ces trois modèles ont chacun leur légitimité dans la simulation numérique de nombreux systèmes électromagnétiques. Il est donc important de construire des méthodes numériques afin de les approcher efficacement. Nous introduirons donc dans le chapitre suivant une méthode générale de résolution des

systemes hyperboliques : la methode de Galerkin Discontinue. Cette methode est particulierement adaptee a la resolution de champs reguliers presentant des discontinuites sur des interfaces stationnaires.

Chapitre 2

Formulation Galerkin Discontinue

Nous avons introduit au chapitre précédent plusieurs modèles d'EDP utiles pour l'électromagnétisme. Ces modèles ont la caractéristique commune de se mettre sous la forme d'un système hyperbolique du premier ordre. Nous allons donc introduire dans ce chapitre une méthode numérique pour approcher ce type de systèmes. Cette méthode est basée sur une formulation faible du problème d'évolution que nous allons d'abord introduire. Nous pourrions ainsi traiter avec le même code plusieurs modèles différents. Avant d'introduire la formulation faible, nous commençons par rappeler de façon très rapide la théorie de Lax-Phillips [50, 60] qui permet de déterminer les conditions aux limites pour lesquelles, le problème d'évolution (1.7 page 11) est bien posé. Puis nous en déduisons une formulation semi-discrète basée sur une approximation polynomiale de la solution à chaque instant. Enfin, nous donnerons une preuve de la convergence de cette formulation semi-discrète.

2.1 Généralités sur les systèmes de Friedrichs

Dans les cas des équations de Maxwell et des équations de Maxwell avec correction de divergence, le système obtenu est donc bien un système de Friedrichs. La théorie générale, dont nous allons rappeler quelques résultats utiles, s'applique également à d'autres systèmes utiles en physique.

Pour que le problème d'évolution soit bien posé dans un domaine spatio-

temporel $\Omega \times [0, T]$, il faut lui adjoindre une condition initiale

$$\mathcal{W}(x, 0) = \mathcal{W}_0(x), \quad x \in \Omega, \quad (2.1)$$

et des conditions aux limites. Nous considérerons des conditions aux limites locales et linéaires. En un point de la frontière $x \in \partial\Omega$ la condition aux limites est décrite par la donnée d'un espace vectoriel $V_b(x)$ et la condition s'écrira

$$\mathcal{W}(x) \in V_b(x). \quad (2.2)$$

Le problème de Cauchy associé au système (1.7 page 11) s'écrit alors

$$\begin{aligned} \partial_t \mathcal{A}^t \mathcal{W} + \Sigma \mathcal{W} + \sum_{i=1}^3 \partial_i \mathcal{A}^i \mathcal{W} &= \mathcal{S}, \text{ sur } \Omega \times [0, T], \\ \mathcal{W}(x) &\in V_b(x), \text{ sur } \partial\Omega \times [0, T], \\ \mathcal{W}(x, 0) &= \mathcal{W}_0(x), \text{ sur } \Omega. \end{aligned} \quad (2.3)$$

L'espace vectoriel $V_b(x)$ doit être choisi de manière à garantir l'existence et l'unicité de la solution. Avant de donner un théorème assurant ces propriétés, définissons une énergie associée à la solution du système (2.3).

Définition 2. *L'énergie \mathcal{E} associée aux champs \mathcal{W} à un instant t est définie par l'intégrale*

$$\mathcal{E} = \int_{\Omega} \frac{1}{2} \mathcal{W} \cdot \mathcal{A}^t \mathcal{W} dx. \quad (2.4)$$

Supposons pour simplifier que $\Sigma = 0$ et $\mathcal{S} = 0$. En multipliant (2.3) par \mathcal{W} et en intégrant par partie, nous obtenons le bilan d'énergie

$$\partial_t \mathcal{E} = - \int_{\partial\Omega} \frac{1}{2} \mathcal{A}^i n_i \mathcal{W} \cdot \mathcal{W} ds. \quad (2.5)$$

Dans le cas d'un second membre nul, la décroissance de l'énergie au cours du temps assure l'unicité de la solution. Nous obtenons donc le résultat suivant :

Théorème 1. *Si*

$$\forall \mathcal{W} \in V_b, \mathcal{A}^i n_i \mathcal{W} \cdot \mathcal{W} \geq 0 \quad (2.6)$$

alors la solution du problème aux limites (2.3) est unique.

L'existence de la solution est un problème mathématiquement beaucoup plus délicat. Les conditions d'existence ont d'abord été analysées par Lax et Philips [50]. Les résultats préliminaires de Lax et Philips ont ensuite été étendus par Rauch [60]. Voir aussi [31, 8, 57]. Nous ne reprendrons pas ces études ici. Mais, en schématisant à l'extrême, l'existence de la solution de (2.3 page précédente) est assurée par l'unicité de la solution d'un problème adjoint avec condition finale qui s'écrit :

$$\begin{aligned} -\partial_t \mathcal{A}^t \mathcal{W} + \Sigma^T \mathcal{W} - \sum_{i=1}^3 \partial_i \mathcal{A}^i \mathcal{W} &= 0, \text{ sur } \Omega \times [0, T], \\ \mathcal{W}(x) &\in V_b^\#(x), \text{ sur } \partial\Omega \times [0, T], \\ \mathcal{W}(x, T) &= \mathcal{W}_T(x), \text{ sur } \Omega. \end{aligned} \quad (2.7)$$

Ce problème adjoint est associé à une condition aux limites adjointe $\mathcal{W} \in V_b^\#$ avec $V_b^\# = (\mathcal{A}^i n_i V_b)^\perp$, *i.e.*

$$\forall \mathcal{W} \in V_b, \forall \mathcal{W}^\# \in V_b^\#, \mathcal{A}^i n_i \mathcal{W} \cdot \mathcal{W}^\# = 0. \quad (2.8)$$

Cette condition implique que la dimension de l'espace V_b doit être égale au nombre de valeurs propres positives ou nulles de $\mathcal{A}^i n_i$ en comptant leur multiplicité. Nous sommes donc amenés à introduire les définitions suivantes des parties positives et négatives de $\mathcal{A}^i n_i$.

Définition 3. Soit M une matrice carrée réelle et diagonalisable sur \mathbb{R} . Soit P une matrice inversible et N une matrice diagonale telles que $M = P^{-1}NP$. Notons $|N|$ la matrice contenant les valeurs absolues des termes de N . La partie positive de M notée M^+ est définie par

$$M^+ = \frac{1}{2} P^{-1} (N + |N|) P. \quad (2.9)$$

La partie négative de M notée M^- est définie par

$$M^- = \frac{1}{2} P^{-1} (N - |N|) P. \quad (2.10)$$

En d'autres termes nous obtenons le résultat suivant

Théorème 2. Le problème aux limites (2.3 page ci-contre) admet une solution unique si et seulement si, en tout point du bord,

$$\forall \mathcal{W} \in V_b, \mathcal{A}^i n_i \mathcal{W} \cdot \mathcal{W} \geq 0 \quad (2.11)$$

et

$$\dim(V_b) = \dim(\ker(\mathcal{A}^i n_i^-)). \quad (2.12)$$

Pour que le problème d'évolution soit bien posé, il faut donc que les conditions aux limites vérifient à la fois la condition de *dissipation* (2.11) et la condition de *maximalité* (2.12).

Remarque 3. *Dans le cas des équations de Maxwell, la matrice $\mathcal{A}^i n_i$ de taille 6×6 a quatre valeurs propres positives ou nulles. La condition de maximalité implique qu'il n'est pas possible d'appliquer plus de deux conditions sur les champs à la limite du domaine. Le système de Maxwell avec correction de divergence permet d'appliquer au système une plus grande variété de conditions aux limites, car aucune de ses valeurs propres ne s'annule.*

2.2 Formulation faible du problème d'évolution

Dans cette section, nous allons construire à partir d'un système de Friedrichs (2.3 page 28) bien posé une formulation faible de type Galerkin Discontinue du problème d'évolution. Le problème est bien posé si les conditions aux limites vérifient les conditions de Lax-Philips (théorème 2). Nous verrons que la condition de Lax-Philips se traduit dans la formulation faible par un choix de flux numérique frontière vérifiant des propriétés très précises de dissipation et de maximalité.

Soit Ω un ouvert de \mathbb{R}^3 . Soit \mathcal{W} une solution du système de lois de conservation (2.3 page 28).

Soit des ouverts $(\Omega_k)_{k=1,\dots,K}$ tels que sur chacun d'eux les paramètres physiques du milieu soient constants et

$$\bigcup_{k=1}^K \overline{\Omega_k} = \overline{\Omega} \text{ et } \forall k \neq k', \Omega_k \cap \Omega_{k'} = \emptyset. \quad (2.13)$$

Nous avons vu au chapitre précédent que les discontinuités stationnaires de la solution vont coïncider avec les discontinuités des matériaux diélectriques. La solution de (2.3 page 28) admet donc des discontinuités uniquement sur l'union des frontières de Ω_k .

Nous ajoutons alors des discontinuités fictives à la solution \mathcal{W} . Pour cela nous définissons un ensemble d'ouverts $\mathcal{M} = (L_i)_{i=1,\dots,N}$ inclus dans Ω tels que

$$\bigcup_{i=1}^N \overline{L_i} = \overline{\Omega} \text{ et } \forall i, \exists k, L_i \subset \Omega_k. \quad (2.14)$$

En particulier, l'union des frontières des Ω_k est incluse dans l'union des frontières des L_i

$$\bigcup_{k=1}^K \partial\Omega_k \subset \bigcup_{i=1}^N \partial L_i. \quad (2.15)$$

Ainsi, la solution \mathscr{W} admet potentiellement des discontinuités sur l'union des frontières des L_i que nous noterons

$$D = \bigcup_{i=1}^N \partial L_i. \quad (2.16)$$

Les ouverts L_i joueront plus tard le rôle d'éléments de maillage. Nous voyons ici la nécessité de faire coïncider les faces des éléments de maillage avec les discontinuités fixes de la solution, *i.e.* avec les discontinuités de matériaux. La solution du problème d'évolution est alors continue sur

$$C = \bigcup_{i=1}^N L_i. \quad (2.17)$$

En d'autres termes, nous supposons que le champ de vecteur $\mathscr{W}(x, t)$ est continument différentiable par rapport au temps et que sa restriction à L_i est continument différentiable par rapport à la variable spatiale sur chaque ensemble \bar{L}_i .

En pratique, des discontinuités en temps des champs électromagnétiques apparaîtraient si la condition initiale est discontinue ou si une source électromagnétique à une forme temporelle de type mesure de Dirac. Une telle discontinuité est difficile à représenter numériquement, car un signal temporel discontinu a un spectre fréquentiel infini. Dans la section 6.2, nous nous intéresserons plus en détail aux limites de validité fréquentielle de la méthode GD d'ordre élevé.

Des discontinuités spatiales apparaissent aux interfaces entre matériaux diélectriques. Ces discontinuités sont fixes dans l'espace et les champs électromagnétiques y présentent une composante normale discontinue. Ainsi, en supposant que les interfaces entre matériaux coïncident avec des frontières d'ouverts L , nous pouvons correctement prendre en compte ces discontinuités.

Nous utilisons la notation $\langle f, \psi \rangle$ pour le crochet distributionnel d'une distribution $f \in \mathcal{D}'(\Omega)$ et d'une fonction test $\psi \in \mathcal{D}(\Omega)$. Rappelons que dans le cas où $f \in L^1_{\text{loc}}$ alors ce crochet vérifie

$$\langle f, \psi \rangle = \int_{\Omega} f \psi dx. \quad (2.18)$$

D'autre part, la dérivée partielle ∂_i de f au sens des distributions est définie par

$$\langle \partial_i f, \psi \rangle := - \langle f, \partial_i \psi \rangle. \quad (2.19)$$

Soit aussi une matrice M définie sur le bord de $\partial\Omega$ qui permet d'appliquer la condition aux limites. Cette matrice est telle que

$$\ker M = V_b \quad (2.20)$$

(voir (2.2 page 28)) pour que

$$M\mathcal{W} = 0 \Leftrightarrow \mathcal{W} \in V_b. \quad (2.21)$$

Supposons alors que nous cherchons une solution $\mathcal{W}(x, t)$ de (2.3 page 28) au sens des distributions. Afin de simplifier l'exposé, nous supposons que la matrice \mathcal{A}^t est égale à l'identité. Nous avons alors

$$\forall \psi \in C^1(\overline{\Omega}), \langle \partial_i \mathcal{W}, \psi \rangle + \langle \mathcal{A}^i \partial_i \mathcal{W}, \psi \rangle + \int_{\partial\Omega} M \mathcal{W} \psi ds = 0. \quad (2.22)$$

\mathcal{W} est discontinue sur $D \setminus \partial\Omega$. Ainsi en appliquant la formule des sauts (voir [65]), nous pouvons écrire l'équation précédente sous la forme

$$\forall \psi \in C^1(\overline{\Omega}), \int_{\Omega} \partial_i \mathcal{W} \psi dx + \int_C \mathcal{A}^i \partial_i \mathcal{W} \psi dx + \int_{D \setminus \partial\Omega} \mathcal{A}^i n_i (\mathcal{W}_R - \mathcal{W}_L) \psi ds + \int_{\partial\Omega} M \mathcal{W} \psi ds = 0. \quad (2.23)$$

L et R représentent les ouverts situés de chaque côté de D . D est orientée de telle sorte que le vecteur normal unitaire à D , noté n , soit orienté de L vers R . Les champs \mathscr{W}_L et \mathscr{W}_R sont les valeurs sur D des champs provenant de chacun des deux côtés L et R .

Maintenant que nous avons établi une formulation faible associée au problème (2.3 page 28), nous pouvons en déduire une formulation Galerkin Discontinue.

2.3 Formulation Galerkin Discontinue

Nous allons maintenant étendre la formulation (2.23 page précédente) à des fonctions tests pouvant également présenter une discontinuité sur les interfaces entre ouverts et ayant la même régularité que la solution. Pour cela, nous introduisons un flux numérique, dépendant des états \mathscr{W}_L et \mathscr{W}_R afin de donner un sens à l'intégrale sur $D \setminus \partial\Omega$ puisque les fonctions tests y sont maintenant discontinues.

Puisque le système de lois de conservation (2.3 page 28) est hyperbolique, la matrice $\mathcal{A}^i n_i$ est diagonalisable dans \mathbb{R} . Soient $\mathcal{A}^i n_i^+$ et $\mathcal{A}^i n_i^-$ les matrices rassemblant respectivement les valeurs propres positives et négatives de la matrice $\mathcal{A}^i n_i$ donnée dans la définition 3.

Dans le cas des équations de Maxwell, la matrice $\mathcal{A}^i n_i$ vaut

$$\mathcal{A}^i n_i = \begin{pmatrix} 0 & -n \times \\ n \times & 0 \end{pmatrix} \text{ où } n \times = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix}. \quad (2.24)$$

Les valeurs propres de cette matrice sont $|n|^2$, $-|n|^2$ et 0. Les matrices $\mathcal{A}^i n_i^+$ et $\mathcal{A}^i n_i^-$ valent

$$\mathcal{A}^i n_i^+ = \frac{1}{2} \begin{pmatrix} -n \times (n \times) & -n \times \\ n \times & -n \times (n \times) \end{pmatrix} \quad (2.25)$$

et

$$\mathcal{A}^i n_i^- = \frac{1}{2} \begin{pmatrix} n \times (n \times) & -n \times \\ n \times & n \times (n \times) \end{pmatrix}. \quad (2.26)$$

en supposant le vecteur n unitaire.

Nous introduisons alors ces matrices dans la formulation faible (2.23 page 32) et obtenons

$$\begin{aligned} \int_{\Omega} \partial_t \mathcal{W} \psi dx + \int_C \mathcal{A}^i \partial_i \mathcal{W} \psi dx + \\ \int_{D \setminus \partial\Omega} \mathcal{A}^i n_i^- (\mathcal{W}_R - \mathcal{W}_L) \psi_L + \mathcal{A}^i n_i^+ (\mathcal{W}_R - \mathcal{W}_L) \psi_R ds + \\ \int_{\partial\Omega} M \mathcal{W}_L \psi_L ds = 0, \end{aligned} \quad (2.27)$$

pour toute fonction test ψ , éventuellement discontinue sur D (voir [47]). Il est important de remarquer ici que cette formulation faible n'a plus de sens dans l'espace des distributions $\mathcal{D}(\Omega)$ puisque les fonctions tests sont maintenant discontinues. Cette formulation a du sens par exemple lorsque \mathcal{W} est de classe C^1 sur toute cellule \bar{L} du maillage \mathcal{M} . Nous pouvons donc introduire l'espace de fonctions suivant :

$$\mathcal{H} = \{f \in L^2(\Omega), \forall L \in \mathcal{M}, f|_L \in C^1(\bar{L})\}. \quad (2.28)$$

Nous cherchons alors la solution \mathcal{W} de (2.27) dans $C^1([0, T], \mathcal{H}^m)$, avec $\mathcal{H}^m = \mathcal{H} \times \cdots \times \mathcal{H}$.

Lorsque les fonctions tests s'annulent en dehors de l'adhérence d'un ouvert L de \mathcal{M} , la formulation GD (2.27) devient

$$\begin{aligned} \int_L \partial_t \mathcal{W} \psi dx + \int_L \mathcal{A}^i \partial_i \mathcal{W} \psi dx + \int_{\partial L \setminus \partial\Omega} \mathcal{A}^i n_i^- (\mathcal{W}_R - \mathcal{W}_L) \psi_L ds + \\ \int_{\partial L \cap \partial\Omega} M \mathcal{W}_L \psi_L ds = 0. \end{aligned} \quad (2.29)$$

En appliquant la formule de Green

$$\int_L \nabla \varphi \psi = - \int_L \nabla \psi \varphi + \int_{\partial L} n \varphi \psi, \quad (2.30)$$

l'équation précédente devient

$$\int_L \partial_t \mathcal{W} \psi dx - \int_L \mathcal{A}^i \partial_i \psi \mathcal{W} dx + \int_{\partial L \setminus \partial \Omega} (\mathcal{A}^i n_i^+ \mathcal{W}_L + \mathcal{A}^i n_i^- \mathcal{W}_R) \psi_L ds + \int_{\partial \Omega} (\mathcal{A}^i n_i + M) \mathcal{W}_L \psi_L ds = 0. \quad (2.31)$$

Nous introduisons alors le flux numérique

$$F(\mathcal{W}_L, \mathcal{W}_R, n) = \mathcal{A}^i n_i^+ \mathcal{W}_L + \mathcal{A}^i n_i^- \mathcal{W}_R \quad (2.32)$$

et le flux de bord

$$F_b(\mathcal{W}, n) = (\mathcal{A}^i n_i + M) \mathcal{W}. \quad (2.33)$$

Le flux numérique est un flux décentré, il est possible de choisir d'autres flux numériques, nous donnerons par la suite quelques exemples de flux couramment utilisés. Nous avons choisi ce flux, car il introduit une dissipation numérique, qui améliore la stabilité du schéma (voir [16, 11, 33]).

Nous obtenons finalement, la formulation GD de notre problème :

trouver $\mathcal{W} \in C^1([0, T], \mathcal{H}^m)$, telle que

$$\forall L \in \mathcal{M}, \forall \psi \in C^1(\bar{L}), \quad \int_L \partial_t \mathcal{W} \psi dx - \int_L F(\mathcal{W}, \mathcal{W}, \nabla \psi) dx + \int_{\partial L \setminus \partial \Omega} F(\mathcal{W}_L, \mathcal{W}_R, n) \psi_L ds + \int_{\partial L \cap \partial \Omega} F_b(\mathcal{W}_L, n) \psi_L ds = 0. \quad (2.34)$$

En ajoutant à cette formulation un état initial $\mathcal{W}(0) = \mathcal{W}_0$, le problème d'évolution ainsi posé admet en principe une solution unique.

Nous avons présenté le formalisme GD avec un flux numérique décentré (2.32). Mais le flux numérique F peut être choisi de différentes manières [16, 11, 33]. Pour obtenir une approximation convergente vers la bonne solution, ce flux doit vérifier les propriétés de consistance

$$F(\mathcal{W}, \mathcal{W}, n) = \mathcal{A}^i n_i \mathcal{W} \quad (2.35)$$

et de conservation

$$F(\mathcal{W}_L, \mathcal{W}_R, -n) = -F(\mathcal{W}_R, \mathcal{W}_L, n). \quad (2.36)$$

Dans le cas des équations de Maxwell, nous pouvons définir une famille de flux numériques plus généraux, qui garantissent la stabilité du schéma numérique.

Un tel flux peut s'exprimer en fonction d'un paramètre $\alpha \geq 0$ (voir [16, 33])

$$F_\alpha(\mathcal{W}_L, \mathcal{W}_R, n) = \begin{pmatrix} -\frac{1}{2}n \times (\mathcal{H}_L + \mathcal{H}_R) + \alpha n \times (n \times (\mathcal{E}_R - \mathcal{E}_L)) \\ \frac{1}{2}n \times (\mathcal{E}_L + \mathcal{E}_R) + \alpha n \times (n \times (\mathcal{H}_R - \mathcal{H}_L)) \end{pmatrix}. \quad (2.37)$$

Le paramètre α définit le décentrage du flux numérique et la dissipation de l'énergie qu'il induit. Nous pouvons citer deux flux spécifiques, le flux décentré obtenu en prenant $\alpha = \frac{1}{2}$ et le flux centré en prenant $\alpha = 0$.

Le décentrage du flux permet de stabiliser le schéma dans certains cas de figure et empêche l'apparition d'ondes parasites. Voir aussi à ce sujet les travaux de Hesthaven [34] ou de Castel, Cohen et Duruffe [11].

Le décentrage de ces flux n'intervient que sur les parties des champs tangentes à n . Il est également possible de définir un flux entraînant un décentrage de champs colinéaire à n . Par exemple, nous pouvons considérer le flux de Rusanov, qui introduit encore plus de stabilité dans le schéma. Ce flux, très dissipatif, est à réserver aux situations particulièrement pathologiques !

$$F_{\text{Rusanov}}(\mathcal{W}_L, \mathcal{W}_R, n) = \begin{pmatrix} \frac{1}{2}n \times (\mathcal{E}_L + \mathcal{E}_R) + \frac{1}{2}\alpha(\mathcal{E}_L - \mathcal{E}_R) \\ -\frac{1}{2}n \times (\mathcal{H}_L + \mathcal{H}_R) + \frac{1}{2}\alpha(\mathcal{H}_L - \mathcal{H}_R) \end{pmatrix}. \quad (2.38)$$

2.4 Stabilité et conditions aux limites

Dans ce qui précède, nous avons introduit une formulation GD d'un système de Friedrichs. Ce système est fermé par une condition initiale et des conditions aux limites. Dans la section 2.1 page 27, nous avons donné un résultat garantissant l'existence et l'unicité de la solution du système de Friedrichs (2.3 page 28). Nous allons maintenant nous intéresser à la stabilité de la méthode GD (2.34 page 35). Nous obtiendrons une propriété devant être vérifiée par la matrice des conditions aux limites.

Plaçons-nous dans le cas où aucune source d'énergie n'est placée dans le domaine. La méthode sera stable si l'énergie du système décroît au cours du temps. Afin d'alléger les écritures, supposons, sans perte de généralité, que la matrice \mathcal{A}^t est égale à l'identité. Dans la définition 2, nous avons donné une énergie associée à la solution \mathcal{W} s'écrivant

$$\partial_t \mathcal{E} = \int_{\Omega} \partial_t \mathcal{W} \cdot \mathcal{W} dx. \quad (2.39)$$

En prenant $\psi = \mathcal{W}$ dans (2.27 page 34), nous pouvons exprimer la dérivée de l'énergie

$$\begin{aligned} \int_{\Omega} \partial_t \mathcal{W} \cdot \mathcal{W} dx &= - \int_C (\mathcal{A}^i \partial_i \mathcal{W}) \cdot \mathcal{W} dx - \int_{\partial\Omega} (M \mathcal{W}_L) \cdot \mathcal{W}_L ds - \\ &\quad \int_{D \setminus \partial\Omega} (\mathcal{A}^i n_i^- (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_L + (\mathcal{A}^i n_i^+ (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_R ds. \end{aligned} \quad (2.40)$$

Nous pouvons écrire

$$\begin{aligned} \int_C (\mathcal{A}^i \partial_i \mathcal{W}) \cdot \mathcal{W} dx &= \int_C \frac{1}{2} \partial_i ((\mathcal{A}^i \mathcal{W}) \cdot \mathcal{W}) dx \\ &= \int_D \frac{1}{2} (\mathcal{A}^i n_i \mathcal{W}_L) \cdot \mathcal{W}_L ds - \int_{D \cap \Omega} \frac{1}{2} (\mathcal{A}^i n_i \mathcal{W}_R) \cdot \mathcal{W}_R ds \\ &= - \int_{D \cap \Omega} \frac{1}{2} (\mathcal{A}^i n_i (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_L ds - \\ &\quad \int_{D \cap \Omega} \frac{1}{2} (\mathcal{A}^i n_i (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_R ds + \\ &\quad \int_{\partial\Omega} \frac{1}{2} (\mathcal{A}^i n_i \mathcal{W}_L) \cdot \mathcal{W}_L ds. \end{aligned} \quad (2.41)$$

De plus,

$$\begin{aligned} \int_{D \cap \Omega} \frac{1}{2} (\mathcal{A}^i n_i (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_L ds - \int_{D \cap \Omega} (\mathcal{A}^i n_i^- (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_L ds \\ = \int_{D \cap \Omega} \frac{1}{2} (|\mathcal{A}^i n_i| (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_L ds \end{aligned} \quad (2.42)$$

et

$$\begin{aligned} & \int_{D \cap \Omega} \frac{1}{2} (\mathcal{A}^i n_i (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_R ds - \int_{D \cap \Omega} (\mathcal{A}^i n_i^+ (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_R ds \\ & = - \int_{D \cap \Omega} \frac{1}{2} (|\mathcal{A}^i n_i| (\mathcal{W}_R - \mathcal{W}_L)) \cdot \mathcal{W}_R ds \end{aligned} \quad (2.43)$$

où $|\mathcal{A}^i n_i| = \mathcal{A}^i n_i^+ - \mathcal{A}^i n_i^-$ est une matrice dont les valeurs propres sont des réels positifs.

Ainsi, nous obtenons une expression de l'évolution de l'énergie ne faisant intervenir que des intégrales surfaciques.

$$\begin{aligned} \int_{\Omega} \partial_t \mathcal{W} \cdot \mathcal{W} dx & = - \int_{D \cap \Omega} \frac{1}{2} (|\mathcal{A}^i n_i| (\mathcal{W}_R - \mathcal{W}_L)) \cdot (\mathcal{W}_R - \mathcal{W}_L) ds \\ & \quad - \int_{\partial \Omega} \left(\left(\frac{1}{2} \mathcal{A}^i n_i + M \right) \mathcal{W}_L \right) \cdot \mathcal{W}_L ds. \end{aligned} \quad (2.44)$$

Cette expression de l'énergie nous permet alors d'énoncer un théorème donnant une condition suffisante de stabilité portant sur les conditions aux limites.

Théorème 3. *L'énergie du système \mathcal{E} décroît si la matrice*

$$\frac{1}{2} \mathcal{A}^i n_i + M \quad (2.45)$$

est positive.

Démonstration. La dérivée en temps de l'énergie vérifie

$$\begin{aligned} \partial_t \mathcal{E} & = - \int_{D \cap \Omega} \frac{1}{2} (|\mathcal{A}^i n_i| (\mathcal{W}_R - \mathcal{W}_L)) \cdot (\mathcal{W}_R - \mathcal{W}_L) ds \\ & \quad - \int_{\partial \Omega} \left(\left(\frac{1}{2} \mathcal{A}^i n_i + M \right) \mathcal{W}_L \right) \cdot \mathcal{W}_L ds. \end{aligned} \quad (2.46)$$

Et donc, puisque la matrice $|\mathcal{A}^i n_i|$ est une matrice positive, si la matrice $\frac{1}{2} \mathcal{A}^i n_i + M$ est aussi positive, l'intégrale

$$\int_{\partial \Omega} \left(\left(\frac{1}{2} \mathcal{A}^i n_i + M \right) \mathcal{W}_L \right) \cdot \mathcal{W}_L ds. \quad (2.47)$$

est positive pour tout $\mathcal{W} \in \mathcal{H}_h^m$.

Dans ce cas la dérivée de \mathcal{E} est négative et donc \mathcal{E} décroît. \square

Remarque 4. *Cette condition de stabilité porte sur l'expression des conditions aux limites utilisées dans la formulation GD. En particulier, ce théorème nous permettra de déduire d'une condition « physique », la (ou les) matrice M la représentant et garantissant la stabilité du schéma en découplant.*

Remarquons également que si la condition de stabilité est vérifiée, alors le théorème 1 assurant l'unicité de la solution est vérifié.

En effet, pour $\mathcal{W} \in \ker(M)$,

$$0 \leq \left(\frac{1}{2}\mathcal{A}^i n_i + M\right)\mathcal{W} \cdot \mathcal{W} = \frac{1}{2}\mathcal{A}^i n_i \mathcal{W} \cdot \mathcal{W}. \quad (2.48)$$

Conclusion :

Nous avons déduit de la formulation faible du problème une formulation Galerkin Discontinue. Cette formulation fait intervenir un flux que nous avons choisi dissipatif. La dissipation du flux permet de stabiliser le schéma et limite l'apparition d'oscillations dans la solution numérique. Cette formulation s'appuie sur une partition de l'espace en ouverts sur lesquels la solution est régulière. Sur les bords de ces ouverts, la solution peut présenter des discontinuités. En faisant coïncider les bords des ouverts avec les discontinuités de paramètres physiques, nous sommes en mesure de représenter correctement les discontinuités de champs y apparaissant. Nous avons enfin donné une condition suffisante de stabilité portant sur les conditions aux limites.

Nous allons maintenant discrétiser la solution suivant les variables d'espace pour obtenir une formulation semi-discrète. Nous donnerons ensuite une preuve de convergence du schéma ainsi obtenu.

2.5 Formulation semi-discrète

Dans les sections précédentes, nous avons introduit la formulation GD (2.34 page 35) d'un système hyperbolique (1.7 page 11). Nous avons supposé que la solution \mathcal{W} de ce système hyperbolique était dans $C^1([0, T], \mathcal{H}^m)$ (voir (2.28 page 34)).

Supposons maintenant que nous avons construit un maillage \mathcal{M} de Ω dont les bords des mailles correspondent exactement aux discontinuités stationnaires de la solution exacte. Nous approximons alors la solution \mathcal{W} du système (1.7 page 11) par une solution discrète W dont les composantes à chaque instant t sont polynomiales dans chaque maille. Supposons de plus que la solution soit approximée dans chaque maille par des fonctions polynomiales de degré d formant un espace vectoriel de degré M engendré par $(\psi_i^L)_{i=0,\dots,M-1}$. Par exemple, cette base pourrait être constituée de produits tensoriels de polynômes de degré au plus d , nous décrirons plus loin un tel espace d'approximation. Notons alors cet espace d'approximation

$$\mathcal{H}_h = \text{vect}\{\psi_i^L, L \in \mathcal{M}, i = 0 \dots M - 1\}. \quad (2.49)$$

Le paramètre h représente un paramètre de finesse du maillage, par exemple le diamètre maximal des mailles. Les fonctions ψ_i^L sont nulles en dehors de la maille L . Elles constituent une base de dimension finie de fonctions définies sur L .

Sur la maille L , nous avons

$$\mathcal{W}(x, t) \simeq W(x, t) = \sum_{j=0}^{M-1} W_j(t) \psi_j^L(x), \quad x \in L. \quad (2.50)$$

Le problème semi-discret s'écrit comme suit.

Trouver $W \in C^1([0, T], \mathcal{H}_h)$ telle que

$$\begin{aligned} \forall L \in \mathcal{M}, \forall \psi \in \mathcal{H}_h, \text{supp}(\psi) \subset \bar{L}, \\ \int_L \partial_t W \psi dx - \int_L F(W, W, \nabla \psi) dx + \\ \int_{\partial L \setminus \partial \Omega} F(W_L, W_R, n) \psi_L ds + \\ \int_{\partial \Omega} F_b(W_L, n) \psi_L ds = 0. \end{aligned} \quad (2.51)$$

Connaissant W à un instant t , le problème (2.51) permet de calculer sa dérivée en temps à ce même instant.

2.6 Convergence du schéma semi-discret

Nous avons maintenant à notre disposition un espace d'approximation et une formulation semi-discrète de notre problème de départ. Nous allons maintenant démontrer la convergence de la solution de (2.51) vers la solution du problème continu (2.3 page 28) pour une condition aux limites radiative. Cette démonstration est tirée de [38] et [47].

Considérons une discrétisation de Ω formée d'ouvert $(L_i)_{i=1,\dots,N}$ deux à deux disjoints et tels que $\bigcup_{i=1}^N \overline{L_i} = \overline{\Omega}$. Supposons construit l'espace d'approximation \mathcal{H}_h (2.49 page ci-contre) formé de fonctions polynomiales par morceaux de degré au plus d .

Soit alors h le diamètre maximal des ouverts $(L_i)_{i=1,\dots,N}$ et notons $D_h = \bigcup \partial L_i$. Comme précédemment, l'ensemble des discontinuités de la solution discrète D_h est muni d'un vecteur normal unitaire n . Ce vecteur est orienté vers l'extérieur de Ω sur l'intersection $\partial\Omega \cap D_h$.

Nous nous intéressons au problème discret (2.51) dans le cas où une condition aux limites radiative, représentée par la matrice

$$M = -\mathcal{A}^i n_i^-, \quad (2.52)$$

est appliquée sur tout le bord du domaine de calcul. Cette condition aux limites sera présentée dans la section 3.4 page 55 et correspond à l'application du flux de bord

$$F_b(W, n) = (\mathcal{A}^i n_i + M)W = \mathcal{A}^i n_i^+ W. \quad (2.53)$$

Dans cette situation, nous pouvons alors énoncer le théorème suivant.

Théorème 4. *La solution du problème semi-discret (2.51 page précédente) converge vers la solution du problème continu (2.3 page 28) lorsque h tend vers zéro.*

Avant de démontrer ce théorème, définissons la forme bilinéaire B définie sur

$\mathcal{H}_h^m \times \mathcal{H}_h^m$ à valeur dans \mathbb{R}

$$\begin{aligned}
B(\varphi, \psi) &= \int_{\Omega} \mathcal{A}^i \partial_i \varphi \cdot \psi \\
&+ \int_{D_h \cap \Omega} \mathcal{A}^i n_i^- (\varphi_R - \varphi_L) \cdot \psi_L \\
&+ \int_{D_h \cap \Omega} \mathcal{A}^i n_i^+ (\varphi_R - \varphi_L) \cdot \psi_R \\
&+ \int_{\partial \Omega} -\mathcal{A}^i n_i^- \varphi_L \cdot \psi_L.
\end{aligned} \tag{2.54}$$

B est une forme bilinéaire (en général non symétrique) positive et en posant $\varphi_R = \psi_R = 0$ sur le bord de Ω

$$B(\varphi, \varphi) = \frac{1}{2} \int_{D_h} |\mathcal{A}^i n_i| (\varphi_R - \varphi_L) \cdot (\varphi_R - \varphi_L). \tag{2.55}$$

La formulation du problème discret peut être mise sous la forme : trouver W dans $C^1([0, T], \mathcal{H}_h^m)$ tel que pour tout ψ_h dans $C^1([0, T], \mathcal{H}_h^m)$

$$\begin{aligned}
\int_{\Omega} \partial_t W \cdot \psi_h + B(W, \psi_h) &= 0, \\
\int_{\Omega} W(x, 0) \cdot \psi_h &= \int_{\Omega} W_0 \cdot \psi_h.
\end{aligned} \tag{2.56}$$

La formulation est équivalente à celle présentée en (2.51 page précédente), mais il est plus pratique pour la démonstration de considérer des fonctions tests ψ_h vectorielles plutôt que scalaires.

Nous pouvons également mettre le problème continu sous la forme :

$$\begin{aligned}
\int_{\Omega} \partial_t \mathcal{W} \cdot \psi + B(\mathcal{W}, \psi) &= 0, \\
\int_{\Omega} \mathcal{W}(x, 0) \cdot \psi &= \int_{\Omega} W_0 \cdot \psi.
\end{aligned} \tag{2.57}$$

Afin de démontrer le théorème de convergence, nous utiliserons le résultat suivant.

Lemme 1. *Il existe une constante positive λ telle que pour tout φ et ψ appartenant à $L^2(D_h)$,*

$$\left| \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i \varphi \cdot \psi \right| \leq \lambda B(\varphi, \varphi)^{1/2} \|\psi\|_{L^2(D_h)}. \quad (2.58)$$

Démonstration. L'inégalité de Cauchy-Schwartz donne la majoration

$$\left| \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i \varphi \cdot \psi \right| \leq \frac{1}{2} \left(\int_{D_h} \mathcal{A}^i n_i \varphi \cdot \mathcal{A}^i n_i \varphi \right)^{1/2} \|\psi\|_{L^2(D_h)}. \quad (2.59)$$

De plus, puisque $\mathcal{A}^i n_i$ est symétrique, elle est diagonalisable en base ortho-normée, *i.e.* $\mathcal{A}^i n_i = P \Lambda P^{-1}$ avec P une matrice unitaire et Λ une matrice diagonale.

Alors,

$$\begin{aligned} \int_{D_h} \mathcal{A}^i n_i \varphi \cdot \mathcal{A}^i n_i \varphi &= \int_{D_h} P \Lambda P^{-1} \varphi \cdot P \Lambda P^{-1} \varphi \\ &= \int_{D_h} \Lambda^2 P^{-1} \varphi \cdot P^{-1} \varphi \\ &\leq |\lambda| \int_{D_h} |\Lambda| P^{-1} \varphi \cdot P^{-1} \varphi \\ &= \lambda \int_{D_h} |\mathcal{A}^i n_i| \varphi \cdot \varphi = \lambda B(\varphi, \varphi). \end{aligned} \quad (2.60)$$

λ désigne la plus grande valeur propre de $\mathcal{A}^i n_i$ en valeur absolue. Cette valeur propre est constante, dans le cas des équations de Maxwell, celle-ci vaut la vitesse de la lumière. \square

Nous pouvons maintenant démontrer le théorème 4.

Démonstration. Notons $\mathcal{V} = \mathcal{H}^m$ l'espace de fonctions dans lequel nous cherchons la solution du problème continu et $\mathcal{V}_h = \mathcal{H}_h^m$ l'espace d'approximation spatiale de la solution.

Notons p la projection L^2 sur l'espace d'approximation \mathcal{V}_h définie par

$$p(\mathcal{W})(x, t) = \sum_{i=0}^{M-1} \mathcal{W}(\tau_L(\hat{x}_i), t) \psi_i^L(x), \text{ pour } x \in L. \quad (2.61)$$

Notons $\Delta = p(\mathcal{W}) - \mathcal{W} \in \mathcal{V}$, l'écart entre la solution continue et sa projection sur \mathcal{V}_h , et $\Delta_h = p(\mathcal{W}) - W \in \mathcal{V}_h$, l'écart entre la projection de \mathcal{W} et la solution discrète.

Remarquons que les écarts Δ et Δ_h vérifient

$$\int_{\Omega} \Delta \cdot \Delta_h = \int_{\Omega} \Delta \cdot \mathcal{A}^i \partial_i \Delta_h = \int_{\Omega} \Delta \cdot \partial_t \Delta_h = 0 \quad (2.62)$$

puisque Δ_h , $\mathcal{A}^i \partial_i \Delta_h$ et $\partial_t \Delta_h$ appartiennent aussi à \mathcal{V}_h et par définition de la projection L^2 . Ainsi

$$0 = \partial_t \int_{\Omega} \Delta \cdot \Delta_h = \int_{\Omega} \Delta \cdot \partial_t \Delta_h + \int_{\Omega} \Delta_h \cdot \partial_t \Delta = \int_{\Omega} \Delta_h \cdot \partial_t \Delta. \quad (2.63)$$

En soustrayant le problème discret au problème continu en prenant la même fonction de base $\phi_h \in \mathcal{V}_h$, nous obtenons

$$\int_{\Omega} \partial_t(\mathcal{W} - W) \cdot \phi_h + B(\mathcal{W} - W, \phi_h) = 0. \quad (2.64)$$

Puis, en prenant comme fonction test Δ_h , nous obtenons

$$\int_{\Omega} \partial_t \Delta_h \cdot \Delta_h + B(\Delta_h, \Delta_h) = \int_{\Omega} \partial_t \Delta \cdot \Delta_h + B(\Delta, \Delta_h) = B(\Delta, \Delta_h). \quad (2.65)$$

La forme bilinéaire B est positive, mais non symétrique, ce qui interdit d'appliquer directement l'inégalité de Cauchy-Schwartz. Nous devons donc

estimer la partie antisymétrique de B .

$$\begin{aligned}
\frac{1}{2}(B(\varphi, \psi) - B(\psi, \varphi)) &= \frac{1}{2} \int_{\Omega} \mathcal{A}^i \partial_i \varphi \cdot \psi - \frac{1}{2} \int_{\Omega} \mathcal{A}^i \partial_i \psi \cdot \varphi \\
&+ \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^- (\varphi_R - \varphi_L) \cdot \psi_L + \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^+ (\varphi_R - \varphi_L) \cdot \psi_R \\
&- \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^- (\psi_R - \psi_L) \cdot \varphi_L - \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^+ (\psi_R - \psi_L) \cdot \varphi_R \\
&= - \int_{\Omega} \mathcal{A}^i \partial_i \varphi \cdot \psi \\
&+ \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^- (\varphi_R - \varphi_L) \cdot \psi_L + \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^+ (\varphi_R - \varphi_L) \cdot \psi_R \\
&- \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^- (\psi_R - \psi_L) \cdot \varphi_L - \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i^+ (\psi_R - \psi_L) \cdot \varphi_R \\
&+ \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i \psi_L \cdot \varphi_L - \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i \psi_R \cdot \varphi_R \\
&= - \int_{\Omega} \mathcal{A}^i \partial_i \varphi \cdot \psi + \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i (\varphi_L + \varphi_R) \cdot (\psi_L - \psi_R)
\end{aligned} \tag{2.66}$$

En appliquant l'inégalité triangulaire et l'inégalité de Cauchy-Schwartz, nous obtenons alors la majoration

$$B(\Delta, \Delta_h) \leq B(\Delta, \Delta)^{1/2} B(\Delta_h, \Delta_h)^{1/2} + \left| \frac{1}{2} \int_{D_h} \mathcal{A}^i n_i (\Delta_L + \Delta_R) \cdot (\Delta_{hL} - \Delta_{hR}) \right|. \tag{2.67}$$

Puis en appliquant le lemme 1, nous obtenons

$$B(\Delta, \Delta_h) \leq B(\Delta_h, \Delta_h)^{1/2} \left(B(\Delta, \Delta)^{1/2} + \lambda \|\Delta\|_{L^2(D_h)} \right). \tag{2.68}$$

De plus, nous disposons des estimations de projection suivantes [61] qui permettent de mesurer l'écart entre \mathscr{W} et son projeté L^2 sur l'espace d'approximation spatiale.

$$\begin{aligned}
\|\Delta\|_{L^2(\Omega)} &\leq Ch^{d+1} \|\mathscr{W}\|_{d+1,m} \\
\|\Delta\|_{L^2(D_h)} &\leq Ch^{d+1/2} \|\mathscr{W}\|_{d+1,m}
\end{aligned} \tag{2.69}$$

où C est une constante ne dépendant que de Ω . Rappelons que h est le diamètre maximal des mailles du maillage. La norme $\|\mathcal{W}\|_{d+1,m}$ est une norme de Sobolev¹ par morceaux :

$$\|\mathcal{W}\|_{d+1,m} = \sum_{i=0}^{N-1} \|\mathcal{W}\|_{H^{d+1}(L_i)}. \quad (2.70)$$

Cette estimation est valable à condition que les mailles ne s'étirent pas lorsque h tend vers 0. C'est le cas par exemple, si pour le passage à la limite sur h , on considère des raffinements uniformes d'un maillage fixé initialement.

Nous obtenons ainsi la majoration,

$$\frac{1}{2} \int_{\Omega} \Delta_h \cdot \Delta_h + B(\Delta_h, \Delta_h) \leq CB(\Delta_h, \Delta_h)^{1/2} h^{d+1/2} \|\mathcal{W}\|_{d+1,m}. \quad (2.71)$$

En appliquant l'inégalité suivante avec $\alpha^2 = \frac{1}{C}$

$$\forall \alpha \neq 0, ab \leq \frac{\alpha^2 a^2}{2} + \frac{b^2}{2\alpha^2} \quad (2.72)$$

et en intégrant de $t = 0$ à T , nous obtenons finalement

$$\begin{aligned} \sqrt{\|p(\mathcal{W}) - W\|_{L^2\Omega}^2 + \int_0^T B(p(\mathcal{W}) - W, p(\mathcal{W}) - W) dt} \\ \leq Ch^{d+1/2} \sqrt{\int_0^T \|\mathcal{W}\|_{d+1,m}^2 dt}. \end{aligned} \quad (2.73)$$

De plus, puisque $\mathcal{W} - W = \Delta_h - \Delta$ et $\Delta = \mathcal{O}(h^{d+1})$, nous obtenons

$$\sqrt{\|\mathcal{W} - W\|_{L^2\Omega}^2} \leq Ch^{d+1/2} \sqrt{\int_0^T \|\mathcal{W}\|_{d+1,m}^2 dt}. \quad (2.74)$$

□

1. La norme de Sobolev est définie par $\|\mathcal{W}\|_{H^{d+1}(L_i)}^2 = \sum_{|\alpha| \leq d+1} \|D^\alpha \mathcal{W}\|_{L^2}^2$.

Remarque 5. *En fait, l'inégalité (2.73 page ci-contre) est plus précise puisqu'elle comprend un contrôle des discontinuités sur les interfaces entre mailles. Ce contrôle montre qu'en un certain sens, le saut de $A^i n_i W$ sur l'ensemble des discontinuités D_h tend vers zéro en norme $L^2(D_h)$ lorsque h tend vers zéro. Comme la longueur de l'ensemble D_h tend vers l'infini, ce contrôle est assez fort. Dans le cas de l'utilisation d'un flux centré, ce terme de contrôle n'est pas présent dans cette estimation. Cela explique pourquoi avec le flux centré, les phénomènes d'oscillations de Gibbs sont en général plus importants.*

Conclusion

En conclusion, nous avons présenté dans ce chapitre une formulation Galerkin Discontinue pouvant être appliquée à divers type de systèmes de Friedrichs. Dans cette formulation intervient un flux numérique. Nous avons choisi un flux décentré introduisant une dissipation et permettant de stabiliser le schéma. Nous avons également donné une propriété s'appliquant aux conditions limites permettant d'assurer la stabilité du schéma.

Nous avons déduit de cette formulation GD une formulation semi-discrète et en avons démontré la convergence en une norme plus forte que la norme L^2 qui rend compte d'un contrôle des discontinuités de la solution.

Nous allons maintenant appliquer la condition de stabilité du théorème 3 afin construire différentes conditions aux limites associées aux équations de Maxwell.

Chapitre 3

Modèles physiques

Les chapitres précédents ont permis de mettre en place un schéma GD semi-discret. Nous avons alors démontré la convergence de ce schéma. Nous avons également énoncé le théorème 3 page 38 qui assure la stabilité du schéma lorsque les conditions aux limites assurent la décroissance de l'énergie associée à la solution. Nous disposons également du théorème 2 page 29 qui assure l'existence et l'unicité de la solution.

Partant de ces propriétés, nous décrirons alors différentes conditions aux limites et surfaciques pouvant être appliquées en conservant la stabilité du schéma dans le cas d'une approximation des équations de Maxwell. Une étude des conditions aux limites admissibles pour les équations de Maxwell est faite dans [7] et pour les équations de Maxwell avec correction de divergence dans [25].

3.1 Condition de conducteur électrique parfait

Cette condition aux limites représente la présence d'un conducteur parfait (PEC) sur le bord du domaine de calcul. Physiquement, cette condition est donnée par

$$n \times \mathcal{E} = 0, \tag{3.1}$$

où n est le vecteur normal unitaire sortant associé au bord du domaine de calcul.

Soit M la matrice associée à cette condition aux limites. Le noyau de M est l'ensemble $\left\{ \mathcal{W} = (\mathcal{E}, \mathcal{H})^T, n \times \mathcal{E} = 0 \right\}$. La condition d'existence (2.12 page 29) de la solution impose que cet ensemble soit de dimension égale à 4, ce qui est le cas ici. L'annulation du champ électrique tangent se compose de deux contraintes dans un espace de dimension 6.

Nous pouvons alors définir une matrice M générique

$$M = \begin{pmatrix} an \times (n \times) + bn \times & 0 \\ cn \times (n \times) + dn \times & 0 \end{pmatrix} \quad (3.2)$$

où a, b, c, d sont des réels non tous nuls.

Pour une telle matrice M , la condition de stabilité du théorème 3 page 38 requiert la positivité de la matrice $\frac{1}{2} \mathcal{A}^i n_i + M$, *i.e.*

$$\begin{aligned} 0 &\leq \int_{\partial\Omega} \left(\left(\frac{1}{2} \mathcal{A}^i n_i + M \right) \mathcal{W}_L \right) \cdot \mathcal{W}_L ds \\ &= \int_{\partial\Omega} \frac{1}{2} (-n \times \mathcal{H}) \cdot \mathcal{E} + (n \times \mathcal{E}) \cdot \mathcal{H} ds \\ &\quad + \int_{\partial\Omega} (an \times (n \times \mathcal{E}) + bn \times \mathcal{E}) \cdot \mathcal{E} ds \\ &\quad + \int_{\partial\Omega} (cn \times (n \times \mathcal{H}) + dn \times \mathcal{H}) \cdot \mathcal{E} ds \\ &= \int_{\partial\Omega} (1 + d) (n \times \mathcal{E}) \cdot \mathcal{H} ds \\ &\quad + \int_{\partial\Omega} c (n \times (n \times \mathcal{H})) \cdot \mathcal{E} ds \\ &\quad + \int_{\partial\Omega} a (n \times (n \times \mathcal{E})) \cdot \mathcal{E} ds. \end{aligned} \quad (3.3)$$

Une condition suffisante pour que cette inégalité soit vraie est

$$\begin{aligned} &\forall \mathcal{E}, \forall \mathcal{H}, \\ &\quad (1 + d)(n \times \mathcal{E}) \cdot \mathcal{H} \geq 0, \\ &\quad c(n \times (n \times \mathcal{H})) \cdot \mathcal{E} \geq 0, \\ &\quad a(n \times (n \times \mathcal{E})) \cdot \mathcal{E} \geq 0. \end{aligned} \quad (3.4)$$

Cette condition est réalisée si $d = -1$, $c = 0$ et $a \leq 0$. La matrice M s'écrit

alors

$$M = \begin{pmatrix} an \times (n \times) + bn \times & 0 \\ -n \times & 0 \end{pmatrix}. \quad (3.5)$$

Avec ces valeurs, la condition de positivité s'écrit

$$0 \leq \int_{\partial\Omega} a (n \times (n \times \mathcal{E})) \cdot \mathcal{E} ds. \quad (3.6)$$

Ainsi, le paramètre a conditionne la dissipation introduite par la condition aux limites. Nous considérerons deux cas particuliers, le cas sans dissipation numérique avec $a = 0$ et $b = 0$

$$M_{\text{pec}} = \begin{pmatrix} 0 & 0 \\ -n \times & 0 \end{pmatrix} \quad (3.7)$$

et un cas introduisant une dissipation correspondant à $a = -1$ et $b = 0$

$$M_{\text{pec ghost}} = \begin{pmatrix} -n \times (n \times) & 0 \\ -n \times & 0 \end{pmatrix}. \quad (3.8)$$

Cette condition aux limites introduit une dissipation numérique qui améliorera la stabilité du schéma dans des cas où l'erreur commise lors de l'intégration numérique devient grande *i.e.* dans des cas où les éléments du maillage sont très déformés. Voir la section 4.3.3 page 85. En outre, cette formulation correspond à la condition aux limites dite de l'« état fantôme », pour laquelle un champ fictif \mathcal{W}^* défini en fonction de \mathcal{W} est imposé à l'extérieur du domaine. Dans le cas du conducteur électrique parfait,

$$\mathcal{W}^* = \begin{pmatrix} \mathcal{E}_\top - \mathcal{E}_\perp \\ \mathcal{H}_\perp - \mathcal{H}_\top \end{pmatrix} \quad (3.9)$$

où les \mathcal{E}_\perp et \mathcal{H}_\perp , respectivement \mathcal{E}_\top et \mathcal{H}_\top , sont les composantes orthogonales, respectivement tangentes, de \mathcal{E} et \mathcal{H} à la surface. Autrement dit, l'« état fantôme » est défini par

$$\mathcal{W}^* = \begin{pmatrix} (n \cdot \mathcal{E})n - n \times (\mathcal{E} \times n) \\ n \times (\mathcal{H} \times n) - (n \cdot \mathcal{H})n \end{pmatrix}. \quad (3.10)$$

Dans ce cas, le flux de bord associé à cette condition aux limites s'écrit

$$F_b(\mathcal{W}_L, n) = F(\mathcal{W}_L, \mathcal{W}^*, n) \quad (3.11)$$

où F est le flux numérique décentré.

Nous comparerons ces deux formes de conditions aux limites dans la section 4.3.3 page 85 en illustrant la stabilisation du schéma apporté par la dissipation présente dans le cas de l'utilisation de la condition aux limites « état fantôme ».

Nous pouvons également utiliser cette condition aux limites pour représenter une plaque parfaitement conductrice constituée de faces d'éléments du maillage. Pour cela, nous introduisons une limite fictive sur les faces des éléments constituant cette surface et y appliquons la condition aux limites définie plus haut.

3.2 Matériau conducteur magnétique parfait

Dans la section 3.1 page 49, nous avons présenté le modèle représentant la condition de conducteur électrique parfait. En suivant le même raisonnement nous pouvons définir une condition aux limites représentant un conducteur magnétique parfait correspondant à

$$n \times \mathcal{H} = 0. \quad (3.12)$$

Nous obtenons alors les matrices correspondantes avec dissipation

$$M_{\text{pmc ghost}} = \begin{pmatrix} 0 & -n \times (n \times) \\ 0 & -n \times \end{pmatrix}. \quad (3.13)$$

et sans dissipation

$$M_{\text{pmc}} = \begin{pmatrix} 0 & 0 \\ 0 & -n \times \end{pmatrix}. \quad (3.14)$$

3.3 Condition aux limites avec impédance de surface (SIBC)

Cette condition aux limites lie les champs électrique et magnétique tangents au bord du domaine, par une relation faisant intervenir une impédance de surface, que nous noterons z , réelle et positive appelée relation de Léontovitch. Cette condition aux limites a été définie par Senior dans [66]. Le choix de l'impédance sera décrit par la suite. Nous pourrions alors construire à partir de cette condition aux limites une condition de type Silver-Müller (voir [2, 1]) ainsi qu'un modèle de plaque conductrice mince appelé modèle de Bérenger [4].

Nous ne considérerons ici que le cas où l'impédance ne dépend pas de la fréquence. Il est possible d'étendre ce modèle à des impédances dépendant de la fréquence comme décrit dans [69] section 10.8 dans le cadre du modèle FDTD.

Sur une telle surface munie d'une normale unitaire sortante n , les champs électromagnétiques vérifient

$$n \times \mathcal{E} + zn \times (n \times \mathcal{H}) = 0. \quad (3.15)$$

En effectuant le produit vectoriel de cette équation par n , nous trouvons une formulation équivalente

$$n \times (n \times \mathcal{E}) - zn \times \mathcal{H} = 0, \quad (3.16)$$

puisque pour tout vecteur $V \in \mathbb{R}^3$,

$$\begin{aligned} n \times (n \times (n \times V)) &= \underbrace{(n \cdot (n \times V))}_{=0} n - \underbrace{(n \cdot n)}_{=1} n \times V \\ &= -n \times V. \end{aligned} \quad (3.17)$$

Nous en déduisons alors une matrice M décrivant cette condition aux limites dépendant de deux paramètres a et b réels non tous deux nuls

$$M = \begin{pmatrix} an \times (n \times) & -azn \times \\ bn \times & bzn \times (n \times) \end{pmatrix}. \quad (3.18)$$

La condition (2.12 page 29) du théorème de Lax-Philips est vérifiée pour tout a et b . Examinons la condition (2.45 page 38) qui requiert la positivité de la matrice $N = \frac{1}{2}\mathcal{A}^i n_i + M$. Soit $\mathcal{W} = (\mathcal{E}, \mathcal{H})^T$ un champ électromagnétique.

$$\begin{aligned} \mathcal{W}^T N \mathcal{W} &= (b + az + 1) \mathcal{H} n \times \mathcal{E} + \\ & a \mathcal{E} \cdot (n \times (n \times \mathcal{E})) + bz \mathcal{H} \cdot (n \times (n \times \mathcal{H})). \end{aligned} \quad (3.19)$$

Or, pour tout vecteur $V \in \mathbb{R}^3$, la forme quadratique qui à V associe $V \cdot (n \times (n \times V))$ est négative, en effet

$$\begin{aligned} V \cdot (n \times (n \times V)) &= V \cdot ((n \cdot V)n - (n \cdot n)V) \\ &= (n \cdot V)^2 - (n \cdot n)(V \cdot V) \leq 0 \end{aligned} \quad (3.20)$$

d'après l'inégalité de Cauchy-Schwartz.

Ainsi, la matrice N est positive si a et b sont négatif et liés par $b + az + 1 = 0$. Ceci équivaut à

$$\frac{-1}{z} \leq a \leq 0 \text{ et } b = -(1 + az). \quad (3.21)$$

Nous remarquons alors que la positivité de l'impédance z est essentielle, dans le cas contraire la condition devant être vérifiée par a ne l'est jamais.

Nous obtenons alors une matrice M représentant cette condition aux limites et assurant l'existence et l'unicité de la solution ainsi que la stabilité du schéma sous la forme

$$M = \begin{pmatrix} an \times (n \times) & -azn \times \\ -(1 + az)n \times & -z(1 + az)n \times (n \times) \end{pmatrix}. \quad (3.22)$$

Le paramètre a définit la dissipation de cette condition aux limites. Lorsqu'il est nul, la matrice N est dégénérée et la condition aux limites ne dissipe pas d'énergie. Au contraire, si $a < 0$ la matrice N est définie positive et la condition aux limites dissipe de l'énergie.

3.4 Condition aux limites de Silver-Müller

Cette condition aux limites se traduit par l'application d'une condition de radiation sur les bords du domaine. Elle a été étudiée dans [2, 1]. Cela se traduit par une condition aux limites SIBC où l'impédance de surface z est égale à l'impédance du vide z_0 , autrement dit les champs électrique et magnétique vérifient

$$n \times \mathcal{E} + z_0 n \times (n \times \mathcal{H}) = 0. \quad (3.23)$$

L'impédance du vide vaut $z_0 = \sqrt{\mu_0/\varepsilon_0}$ et $z_0 = 1$ pour la version adimensionnée des équations de Maxwell. Considérons le cas adimensionné. Dans la section 3.3 page 53 nous avons obtenu une matrice représentant cette condition aux limites dépendant d'un paramètre de dissipation vérifiant dans le cas présent $-1 \leq a \leq 0$

$$M = \begin{pmatrix} an \times (n \times) & -an \times \\ -(1+a)n \times & -(1+a)n \times (n \times) \end{pmatrix}. \quad (3.24)$$

En prenant le paramètre a égale à $-\frac{1}{2}$, nous obtenons une matrice $M = -\mathcal{A}^i n_i^-$. Dans ce cas, le flux associé à cette condition aux limites vaut

$$F_{SM}(\mathcal{W}, n) = (\mathcal{A}^i n_i + M) \mathcal{W} = \mathcal{A}^i n_i^+ \mathcal{W}. \quad (3.25)$$

Considérons maintenant une condition de Silver-Müller inhomogène. Cette condition aux limites permet d'injecter dans le domaine de calcul un champ incident que nous noterons \mathcal{W}_i . Cette condition se traduit par

$$M(\mathcal{W} - \mathcal{W}_i) = 0. \quad (3.26)$$

Le flux associé à cette condition inhomogène s'exprime en fonction du flux obtenu dans le cas homogène

$$\begin{aligned} F_{SMI}(\mathcal{W}, n) &= F_{SM}(\mathcal{W}, n) - M\mathcal{W}_i \\ &= \mathcal{A}^i n_i^+ \mathcal{W} + \mathcal{A}^i n_i^- \mathcal{W}_i \\ &= F(\mathcal{W}, \mathcal{W}_i, n). \end{aligned} \quad (3.27)$$

Nous retrouvons ainsi le flux numérique décentré F .

3.5 Modèle de Bérenger

Afin de simuler la présence d'une plaque de faible épaisseur constituée d'un matériau conducteur, nous remplaçons le flux numérique standard par un flux spécifique dépendant de la conductivité et de l'épaisseur de la plaque. Ce modèle permet de substituer une surface à un volume dont le maillage contraindrait fortement le pas de temps du schéma. Ce modèle a été introduit par Bérenger dans [4].

Soit S une telle surface orientée par sa normale unitaire n , notons \mathcal{E}_{T-} , respectivement \mathcal{E}_{T+} , et \mathcal{H}_{T-} , respectivement \mathcal{H}_{T+} , les parties tangentes des champs électrique et magnétique du côté opposé à la normale, respectivement pointé par la normale. Soit de plus, z_s et z_t les impédances de surface et de transfert de la plaque. Les champs électrique et magnétique tangents vérifient la relation de Léontovitch

$$\begin{aligned}\mathcal{E}_{T-} &= -z_s n \times \mathcal{H}_{T-} - z_t n \times \mathcal{H}_{T+}, \\ \mathcal{E}_{T+} &= z_s n \times \mathcal{H}_{T+} + z_t n \times \mathcal{H}_{T-}.\end{aligned}\tag{3.28}$$

En prenant pour hypothèse que l'épaisseur de peau est supérieure à l'épaisseur de la plaque, nous pouvons approcher les impédances de surface et de transfert par

$$z_s = z_t = \frac{1}{\sigma d}\tag{3.29}$$

pour une plaque d'épaisseur d et de conductivité σ .

En terme de fréquence, cette approximation reste valide pour des fréquences inférieures à la fréquence de coupure de la plaque

$$f_c = \frac{1}{2\pi\mu\sigma d^2}.\tag{3.30}$$

Au-delà de la fréquence de coupure, les impédances de surface et de transfert ont une dépendance forte à la fréquence.

Considérons dans un premier temps une plaque placée sur un bord du domaine de calcul et munie d'une normale orientée vers l'extérieur. En remarquant que la partie tangente d'un vecteur $V \in \mathbb{R}^3$ s'exprime $V_T =$

$-n \times (n \times V)$, la relation de Léontovitch s'écrit

$$n \times (n \times \mathcal{E}) - z_s n \times \mathcal{H} = 0. \quad (3.31)$$

Nous retrouvons alors la condition aux limites SIBC et nous en déduisons la matrice M représentant la condition présente

$$M = \begin{pmatrix} an \times (n \times) & -az_s n \times \\ -(1 + az_s)n \times & -z_s(1 + az_s)n \times (n \times) \end{pmatrix}. \quad (3.32)$$

La condition inhomogène s'écrit alors, pour un champ incident $\mathcal{W}_i = (\mathcal{E}_i, \mathcal{H}_i)^T$,

$$\begin{aligned} 0 &= M(\mathcal{W} - \mathcal{W}_i) \\ &= \begin{pmatrix} a(n \times (n \times (\mathcal{E} - \mathcal{E}_i)) - z_s n \times (\mathcal{H} - \mathcal{H}_i)) \\ -(1 + az_s)(n \times (\mathcal{E} - \mathcal{E}_i) + z_s n \times (n \times (\mathcal{H} - \mathcal{H}_i))) \end{pmatrix}. \end{aligned} \quad (3.33)$$

En prenant $\mathcal{W}_i = (0, \mathcal{H}_+)^T$, nous retrouvons la relation de Léontovitch. Nous pouvons alors choisir le paramètre a assurant la stabilité du schéma, *i.e.* $-1/z_s \leq a \leq 0$. Par exemple, en prenant $a = 0$, nous obtenons un flux non dissipatif associé au modèle de Bérenger s'exprimant par

$$\begin{aligned} F_{\text{Bérenger}}(\mathcal{W}_L, \mathcal{W}_R, n) &= (\mathcal{A}^i n_i + M)\mathcal{W}_L - M\mathcal{W}_R \\ &= \begin{pmatrix} -n \times \mathcal{H}_L \\ zn \times (n \times (\mathcal{H}_R - \mathcal{H}_L)) \end{pmatrix}. \end{aligned} \quad (3.34)$$

Il est également possible d'ajouter une dissipation dans un but de stabilisation du schéma en prenant a non nul.

3.6 Injection d'un champ incident

Dans cette section, nous décrivons une condition surfacique permettant l'injection d'un champ incident dans le volume de calcul. Une description de ce type de source électromagnétique dans le cadre de la méthode des différences finies est décrite dans [69] chap.5.

Définissons dans un premier temps, les solutions dites d'ondes planes des équations de Maxwell.

Proposition 7. Soit $g : [0, T] \rightarrow \mathbb{R}$ une fonction \mathcal{C}^1 définissant l'amplitude de l'onde plane au cours du temps. Soit E_0 , H_0 et V trois vecteurs de \mathbb{R}^3 formant une base orthogonale directe telle que $\|V\| = 1$ et $\|H_0\| = \sqrt{\varepsilon/\mu}\|E_0\|$. Les champs électrique et magnétique

$$\begin{aligned}\mathcal{E}(t, X) &= E_0 g(t - c^{-1}V \cdot X) \\ \mathcal{H}(t, X) &= H_0 g(t - c^{-1}V \cdot X)\end{aligned}\tag{3.35}$$

sont solutions des équations de Maxwell (1.1 page 8) dans le cas où le courant \mathcal{J} et les conductivités σ et σ^* sont nuls.

Démonstration. On a $H_0 \times V = \sqrt{\frac{\varepsilon}{\mu}}E_0$. Considérons l'équation de Maxwell-Ampère (1.1a page 8),

$$\begin{aligned}\varepsilon \partial_t \mathcal{E} - \nabla \times \mathcal{H} &= \varepsilon E_0 g'(t - c^{-1}V \cdot X) - \nabla g(t - c^{-1}V \cdot X) \times H_0 \\ &= \varepsilon E_0 g'(t - c^{-1}V \cdot X) + c^{-1}V \times H_0 g'(t - c^{-1}V \cdot X) \\ &= \varepsilon E_0 g'(t - c^{-1}V \cdot X) - c^{-1} \sqrt{\frac{\varepsilon}{\mu}} E_0 g'(t - c^{-1}V \cdot X) \\ &= \left(\varepsilon - c^{-1} \sqrt{\frac{\varepsilon}{\mu}} \right) E_0 g'(t - c^{-1}V \cdot X) \\ &= 0.\end{aligned}\tag{3.36}$$

De la même manière, l'équation de Maxwell-Faraday (1.1b page 8) est vérifiée par \mathcal{E} et \mathcal{H} . \square

Soit \mathcal{S} une surface fermée, incluse dans le domaine de calcul Ω . Nous appellerons zone de champ total le volume délimité par cette surface et zone de champ diffracté son complémentaire. La surface qui sépare les deux zones est appelée surface de Huygens.

Nous allons définir un flux numérique appliquée sur cette surface et permettant d'injecter une onde plane dans la partie champ totale du domaine et laissant sortir les ondes diffractées. En appliquant le principe de superposition, nous définissons deux flux s'appliquant de part et d'autre de la surface de Huygens

$$\begin{aligned}F_{in}(\mathcal{W}_L, \mathcal{W}_R, n) &= F(\mathcal{W}_L, \mathcal{W}_R + \mathcal{W}_i, n), \\ F_{out}(\mathcal{W}_L, \mathcal{W}_R, n) &= F(\mathcal{W}_L, \mathcal{W}_R - \mathcal{W}_i, n)\end{aligned}\tag{3.37}$$

où F est le flux numérique du schéma. Dans cette approche, nous supposons de plus que le champ incident est nul dans la partie champ total du domaine à l'instant initial.

Nous pouvons également utiliser ce modèle de surface rayonnante afin d'injecter un champ incident dans une structure de type guide d'onde. Pour cela, une surface de Huygens est placée sur le port d'entrée du guide. Dans ce cas nous n'avons pas à proprement parler de zone de champ total et de zone de champ diffracté. De plus, la propagation des ondes électromagnétiques dans un guide est fortement liée à la forme de ce guide.

Par exemple dans un guide rectangulaire, les champs se structurent sous la forme de mode. Ces modes peuvent être propagatifs ou évanescents. Dans le cas d'un guide rectangulaire, les modes propagatifs sont classés suivant l'allure transverse des champs électromagnétiques qui suit une répartition sinusoïdale. Ces modes sont repérés par deux entiers, on parlera de mode transverse électrique TE_{mn} ou transverse magnétique TM_{mn} . Cette problématique est abordée dans [17]. Chaque mode est lié à une fréquence, ainsi l'excitation d'un ou plusieurs modes précis d'un guide est difficile.

En utilisant une surface de Huygens pour alimenter un guide nous allons provoquer l'apparition d'ondes parasites dues à la non-adaptation du champ incident à la géométrie du guide. Toutefois, ces ondes sont évanescentes et après une courte distance ne perdurent que des ondes propagatives. Par exemple, lors de l'excitation d'un guide rectangulaire, cette source permet de propager dans le guide des champs électromagnétiques dont le mode dominant est une mode TE_{10} (ou TE_{01} suivant l'orientation du champ électrique).

Conclusion

Nous avons défini plusieurs modèles de matériaux représentés par des conditions aux limites ou surfaciques associées aux équations de Maxwell. Ces conditions aux limites sont représentées par des matrices M vérifiant les théorèmes 2 page 29 et 3 page 38. Ainsi, ces conditions aux limites assurent l'existence et l'unicité de la solution, ainsi que la stabilité du schéma semi-discret. Nous avons également muni certaines conditions aux limites d'un paramètre de dissipation. Cette dissipation nous permet de stabiliser le schéma numérique.

Nous allons maintenant décrire une discrétisation des champs électromagnétiques sur maillage hexaédrique et donnerons l'expression des intégrales apparaissant dans la formulation GD. En ajoutant au schéma semi-discret un

schéma d'intégration en temps, nous obtiendrons un schéma numérique GD pouvant est programmé. Nous discuterons également la validité des formules de quadratures pour cette discrétisation, ainsi que la stabilité du schéma discret.

Chapitre 4

Interpolation Galerkin discontinue

La méthode Galerkin Discontinue (GD) peut être théoriquement appliquée à tout type de maillage. Nous étudierons ici le cas où les éléments du maillage (ou mailles) sont des hexaèdres ayant des arêtes droites formant un maillage conforme de Ω . La méthode GD appliquée à un maillage tétraédrique a été largement étudiée, on peut citer par exemple les travaux de [34] et [24]. L'utilisation des hexaèdres pour GD a été étudiée par exemple par [16]. Les hexaèdres permettent diverses optimisations de la formulation GD et s'adaptent bien à la parallélisation de la méthode sur GPU.

L'espace d'approximation dans lequel nous chercherons la solution numérique de notre problème est engendré par des fonctions de base dont le support est inclus dans une unique maille. Là aussi, il est possible de choisir divers types de fonctions de base [44, 9]. Dans cette thèse, nous nous concentrerons sur des bases de polynômes dites nodales [34, 16]. Ces bases sont constituées de produits tensoriels de polynômes d'interpolation de Lagrange. Les points d'interpolation choisis sont les points d'intégration de Gauss-Legendre. Ce choix de polynômes et de points d'interpolation permet plusieurs optimisations qui vont rendre la méthode GD particulièrement efficace.

4.1 Espace d'approximation

Dans un premier temps, nous construisons l'espace d'approximation dans lequel nous cherchons la solution numérique de notre problème. La construction de cet espace s'appuie sur la définition d'un élément de référence, ici

le cube unité. Une transformation géométrique nous permet alors de passer du cube de référence à une maille hexaédrique. Les fonctions de base sont alors constituées de compositions entre cette transformation et des fonctions polynomiales définies sur le cube unité. Ces fonctions polynomiales sont des produits tensoriels de polynômes de Lagrange associés aux points de Gauss-Legendre. Ces points nous serviront également à définir des formules de quadrature permettant d'estimer les intégrales volumiques et surfaciques apparaissant dans la méthode GD. Ces choix de fonctions de base et de formule de quadrature permettent diverses simplifications de la formulation GD discrète.

4.1.1 Maille de référence

Nous choisissons comme élément de référence le cube unité que nous noterons $\hat{H} = [0, 1]^3$. Nous pouvons alors définir une transformation géométrique naturelle entre cet élément et un hexaèdre quelconque, ce point sera abordé au paragraphe 4.1.2 page 69. Nous commençons par définir des points et poids d'intégration volumiques dans cet élément, puis des points et poids d'intégration surfaciques sur ces faces. Nous définirons ensuite des fonctions de base en nous appuyant sur les points volumiques.

Soit d un entier positif. Nous appellerons par la suite cet entier l'ordre d'approximation du schéma.

Remarque 6. *Dans la présentation de notre choix d'interpolation, nous ferons commencer les numérotations des noeuds, fonctions de base, etc. à partir de l'indice 0. Ce choix permet certaines simplifications d'écriture et correspond aussi à la programmation en C ou C++ de la méthode GD que nous avons réalisée.*

Points d'intégration et formules de quadratures

Soient donc $(\lambda_p)_{p=0,\dots,d}$ et $(\xi_p)_{p=0,\dots,d}$ les poids et points de quadrature 1D de Gauss-Legendre de degré d sur l'intervalle $[0, 1]$. Nous allons passer de la dimension 1 à la dimension 3 par des techniques de produits tensoriels. À tout multi-indice $(p(0), p(1), p(2))$ d'entiers compris entre 0 et d nous pouvons associer de manière bijective un unique entier $i = p(0) + (d+1)p(1) + (d+1)p(2)$ compris entre 0 et $(d+1)^3 - 1$. Des divisions euclidiennes permettent de retrouver $(p(0), p(1), p(2))$ de manière unique à partir de i .

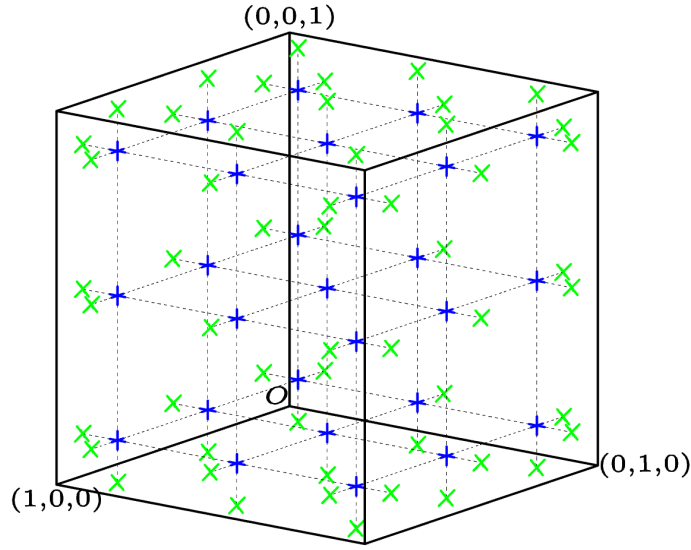


FIGURE 4.1 – Points d’intégration volumiques (« + » bleu) et surfaciques (« × » vert) présents dans l’élément de référence $[0,1]^3$ pour $d = 2$. Les alignements entre ces points sont représentés par les lignes pointillées.

Nous pouvons alors définir $(d + 1)^3$ points d’intégration volumiques

$$\hat{x}_i = (\xi_{p(0)}, \xi_{p(1)}, \xi_{p(2)}) \in \hat{H} \text{ où } i = p(0) + (d + 1)(p(1) + (d + 1)p(2)) \quad (4.1)$$

et $(d + 1)^3$ poids d’intégration associés

$$\omega_i = \lambda_{p(0)}\lambda_{p(1)}\lambda_{p(2)} \text{ où } i = p(0) + (d + 1)(p(2) + (d + 1)p(3)). \quad (4.2)$$

Comme dit plus haut, les points et poids d’intégration sont numérotés de 0 à $(d + 1)^3 - 1$. Cette numérotation permet de retrouver plus facilement les numéros des points et poids unidimensionnels à partir de divisions euclidiennes. Les points d’intégration volumiques sont représentés par des « + » bleus sur la figure FIGURE 4.1 dans le cas où $d = 2$.

Nous approcherons alors les intégrales sur le cube unité \hat{H} par la formule de

quadrature 3D découlant directement de la formule de quadrature sur $[0, 1]$

$$\int_{\hat{H}} \psi d\hat{x} \simeq \sum_{i=0}^{(d+1)^3-1} \omega_i \psi(\hat{x}_i). \quad (4.3)$$

Grâce aux propriétés des quadratures de Gauss-Legendre, nous savons que

Proposition 8. *La formule de quadrature associée aux points (\hat{x}_i) et poids (ω_i) est exacte pour des produits tensoriels de polynômes de degrés au plus $2d + 1$.*

Ajoutons à ces points d'intégration, des points d'intégration situés sur les faces du cube \hat{H} . Notons $(\hat{Q}_f)_{f=0,\dots,5}$ les 6 faces quadrangulaires du cube \hat{H} . Nous avons choisi la numérotation suivante des faces du cube de référence

$$\begin{aligned} \hat{Q}_0 &= [0, 1] \times \{0\} \times [0, 1], \\ \hat{Q}_1 &= \{1\} \times [0, 1] \times [0, 1], \\ \hat{Q}_2 &= [0, 1] \times \{1\} \times [0, 1], \\ \hat{Q}_3 &= \{0\} \times [0, 1] \times [0, 1], \\ \hat{Q}_4 &= [0, 1] \times [0, 1] \times \{0\}, \\ \hat{Q}_5 &= [0, 1] \times [0, 1] \times \{1\}. \end{aligned} \quad (4.4)$$

Nous définissons sur la face \hat{Q}_f , $(d + 1)^2$ points d'intégration notés

$$(\hat{x}_{f,i'}). \quad (4.5)$$

Ces points sont numérotés de 0 à $(d + 1)^2$ sur chaque face. Comme précédemment cette numérotation permet de retrouver facilement les indices des points unidimensionnels à partir de divisions euclidiennes.

Ces points sont les projetés sur les faces des points d'intégration dans le volume. Ils sont symbolisés sur la figure FIGURE 4.1 page précédente par les croix (\times) vertes dans le cas où $d = 2$. Afin de les définir rigoureusement, nous commençons par considérer le tableau suivant, donnant pour chaque face les directions des deux axes tangents à la face puis de l'axe normal à la face. La dernière colonne du tableau correspond à la valeur sur la face de la

dernière coordonnée :

$$\sigma = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 1 & 2 & 0 & 1 \\ 2 & 0 & 1 & 1 \\ 2 & 1 & 0 & 0 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 2 & 1 \end{pmatrix}. \quad (4.6)$$

Par exemple, pour la face $f = 0$ (première ligne du tableau), les coordonnées tangentes correspondent aux axes \hat{x}^0, \hat{x}^2 , la direction normale correspond à l'axe \hat{x}^1 . Enfin, pour cette face la coordonnée normale $\hat{x}^1 = 0$. Considérons maintenant un indice i' compris entre 0 et $(d+1)^2 - 1$ d'un point de face. Nous pouvons trouver de façon unique deux indices $p'(0)$ et $p'(1)$ compris entre 0 et d tels que

$$i' = p'(0) + (d+1)p'(1). \quad (4.7)$$

Alors le point correspondant sur la face f est donné par

$$\hat{x}_{f,i'} = (\hat{x}^0, \hat{x}^1, \hat{x}^2) \quad (4.8)$$

avec

$$\hat{x}^{\sigma(f,0)} = \xi_{p'(0)}, \quad \hat{x}^{\sigma(f,1)} = \xi_{p'(1)}, \quad \hat{x}^{\sigma(f,2)} = \sigma(f,3). \quad (4.9)$$

À ce point d'intégration correspond un poids d'intégration calculé à partir des poids d'intégration 1D

$$\omega_{f,i'} = \lambda_{p'(0)} \lambda_{p'(1)}. \quad (4.10)$$

Ces points et poids d'intégration définissent une formule de quadrature permettant d'approcher les intégrales sur les faces de l'élément \hat{H} .

$$\int_{\hat{Q}_f} \psi \simeq \sum_{i'=0}^{(d+1)^2-1} \omega_{f,i'} \psi(\hat{x}_{f,i'}). \quad (4.11)$$

Proposition 9. *La formule de quadrature (4.11 page précédente) est exacte pour des produits tensoriels de polynômes de degré au plus $2d + 1$.*

Nous pouvons aussi définir le projeté sur la face \hat{Q}_f d'un point d'intégration volumique \hat{x}_i avec

$$i = p(0) + (d + 1)(p(1) + (d + 1)p(2)). \quad (4.12)$$

C'est un point $\hat{x}_{f,i'}$ que l'on peut calculer à partir d'une application π calculant l'indice i' en fonction de l'indice f de la face et de l'indice du point volumique i

$$\pi(f, i) = p(\sigma(f, 0)) + (d + 1)p(\sigma(f, 1)). \quad (4.13)$$

Pour finir, nous munissons les faces de l'élément de référence d'une normale unitaire sortante notée \hat{n} valant

$$\hat{n} = \begin{cases} (0, -1, 0)^T & \text{sur la face 0} \\ (1, 0, 0)^T & \text{sur la face 1} \\ (0, 1, 0)^T & \text{sur la face 2} \\ (-1, 0, 0)^T & \text{sur la face 3} \\ (0, 0, -1)^T & \text{sur la face 4} \\ (0, 0, 1)^T & \text{sur la face 5} \end{cases} \quad (4.14)$$

Les normales aux faces sont représentées sur la figure FIGURE 4.2 page ci-contre.

Fonctions de base

Considérons maintenant $(\ell_p)_{p=0,\dots,d}$ les polynômes de Lagrange associés aux points $(\xi_p)_{p=0,\dots,d}$.

Ces polynômes sont donnés par

$$\ell_p(\xi) = \prod_{\substack{q=0 \\ q \neq p}}^d \frac{\xi - \xi_q}{\xi_p - \xi_q} \quad (4.15)$$

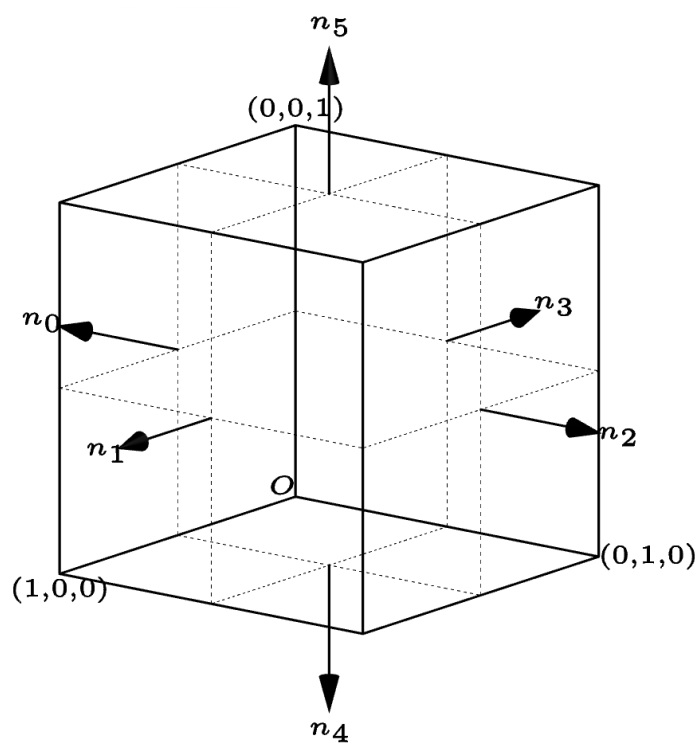


FIGURE 4.2 – Orientations des normales des faces de l'élément de référence.

et vérifient

$$\ell_p(\xi_q) = \delta_{p,q}, \quad (4.16)$$

où $\delta_{p,q}$ est le symbole de Kronecker qui vaut 1 si $p = q$ et 0 sinon.

Nous définissons la base d'approximation sur \hat{H} comme étant formée des produits tensoriels des polynômes ℓ_p . Plus précisément, ces fonctions de base sont données par

$$\hat{\psi}_i : \begin{array}{ccc} \hat{H} & \longrightarrow & \mathbb{R} \\ (\hat{x}^0, \hat{x}^1, \hat{x}^2) & \longmapsto & \ell_{p(0)}(\hat{x}^0)\ell_{p(1)}(\hat{x}^1)\ell_{p(2)}(\hat{x}^2) \end{array} \quad (4.17)$$

où $i = p(0) + (d+1)(p(1) + (d+1)p(2))$.

Proposition 10. *La formule de quadrature (4.3 page 64) est exacte pour des combinaisons linéaires de produits de fonctions de base décrites plus haut. De plus,*

$$\int_{\hat{H}} \sum_{i=0}^{(d+1)^3-1} \sum_{j=0}^{(d+1)^3-1} a_{i,j} \hat{\psi}_i \hat{\psi}_j dx = \sum_{i=0}^{(d+1)^3-1} a_{i,i} \omega_i. \quad (4.18)$$

Remarque 7. *La formule de quadrature (4.3 page 64) est exacte, car nous utilisons des points de Gauss-Legendre. Si nous utilisons des points de Gauss-Lobatto, La proposition précédente ne serait plus valide.*

La proposition suivante nous permettra de simplifier l'extrapolation des champs sur les points d'intégration surfaciques. Elle nous permet de trouver le sous-ensemble des fonctions de base nécessaire à cette extrapolation.

Proposition 11. *Une fonction de base donnée est nulle en tous les points d'intégration sur les faces, excepté celui correspondant au projeté du point volumique correspondant à cette fonction de base. Autrement dit*

$$\hat{\psi}_i(\hat{x}_{f,i'}) = \begin{cases} \ell_{p(\sigma(f,2))}(\sigma(f,3)) & , \text{ si } i' = \pi(f,i) \\ 0 & , \text{ sinon.} \end{cases} \quad (4.19)$$

Démonstration. Soit $\hat{\psi}_i$ une fonction de base et $\hat{x}_i = (\xi_{p(0)}, \xi_{p(1)}, \xi_{p(2)})$ le point d'interpolation correspondant.

Soit $\hat{x}_{f,i'}$ un point d'intégration sur la face f avec $i' = p'(0) + (d+1)p'(1)$.

Alors, en ce point la fonction de base vaut

$$\begin{aligned}
\hat{\psi}_i(\hat{x}_{f,i'}) &= \ell_{p(\sigma(f,0))}(\xi_{p'(0)})\ell_{p(\sigma(f,1))}(\xi_{p'(1)})\ell_{p(\sigma(f,2))}(\sigma(f,3)) \\
&= \delta_{p(\sigma(f,0))n,p'(0)}\delta_{p(\sigma(f,1))n,p'(1)}\ell_{p(\sigma(f,2))}(\sigma(f,3)) \\
&= \delta_{\pi(f,i),i'}\ell_{p(\sigma(f,2))}(\sigma(f,3))
\end{aligned} \tag{4.20}$$

□

Nous pouvons maintenant définir la transformation géométrique permettant de passer du cube unité à une maille hexaédrique.

4.1.2 Transformation géométrique

Soit $(\hat{S}_i)_{i=0,\dots,7}$ les huit sommets du cube unité \hat{H} donnés par

$$\begin{aligned}
\hat{S}_0 &= (0, 0, 0), \\
\hat{S}_1 &= (1, 0, 0), \\
\hat{S}_2 &= (0, 1, 0), \\
\hat{S}_3 &= (1, 1, 0), \\
\hat{S}_4 &= (0, 0, 1), \\
\hat{S}_5 &= (1, 0, 1), \\
\hat{S}_6 &= (0, 1, 1), \\
\hat{S}_7 &= (1, 1, 1).
\end{aligned} \tag{4.21}$$

Définissons les huit fonctions de forme associées à chacun de ces sommets

$$\begin{aligned}
\phi_0(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= (1 - \hat{x}^0)(1 - \hat{x}^1)(1 - \hat{x}^2), \\
\phi_1(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= \hat{x}^0(1 - \hat{x}^1)(1 - \hat{x}^2), \\
\phi_2(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= (1 - \hat{x}^0)\hat{x}^1(1 - \hat{x}^2), \\
\phi_3(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= \hat{x}^0\hat{x}^1(1 - \hat{x}^2), \\
\phi_4(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= (1 - \hat{x}^0)(1 - \hat{x}^1)\hat{x}^2, \\
\phi_5(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= \hat{x}^0(1 - \hat{x}^1)\hat{x}^2, \\
\phi_6(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= (1 - \hat{x}^0)\hat{x}^1\hat{x}^2, \\
\phi_7(\hat{x}^0, \hat{x}^1, \hat{x}^2) &= \hat{x}^0\hat{x}^1\hat{x}^2.
\end{aligned} \tag{4.22}$$

Chacune des ces fonctions de forme prend une valeur nulle sur chacun des sommets du cube unité excepté au point lui correspondant, *i.e.*

$$\phi_i(\hat{S}_j) = \delta_{i,j}. \tag{4.23}$$

Soit L un hexaèdre appartenant au maillage \mathcal{M} . Il est défini par ses huit sommets $(S_i)_{i=0,\dots,7}$. Nous appellerons point physique et noterons x un point situé dans la maille L et nous appellerons point de référence et noterons \hat{x} un point de la maille \hat{H} .

Nous définissons alors la transformation géométrique τ_L dont l'image du cube unité est l'hexaèdre s'appuyant sur les huit sommets $(S_i)_{i=0,\dots,7}$. Cette transformation est donnée par la formule classique, faisant intervenir les fonctions de forme,

$$\tau_L(\hat{x}) = \sum_{i=0}^7 \phi_i(\hat{x})S_i. \tag{4.24}$$

Nous supposons que cette transformation est une bijection entre \hat{H} et L , et que son jacobien

$$\text{jac}(\tau_L) = \det(\tau_L') \tag{4.25}$$

est strictement positif en tout point de \hat{H} . Les noeuds S_i et la connectivité maille vers noeuds sont en général fournis par un logiciel de maillage (ou mailleur). Contrairement aux maillages en tétraèdres, il peut arriver que

le mailleur produise exceptionnellement des éléments dégénérés, c'est-à-dire que le jacobien de τ_L s'annule en certains points de L . Il est important de vérifier, avant de lancer un calcul, que le maillage ne présente pas des éléments dégénérés. Pour cela, nous pouvons calculer le jacobien $\text{jac}(\tau_L)$ aux points d'intégration et vérifier qu'il ne s'annule pas. Nous pouvons également citer la méthode présentée dans [37] et implémentée dans l'outil de maillage Gmsh.

Dans ce qui suit, nous supposons que toutes les mailles considérées ne sont pas dégénérées et que la numérotation de leurs sommets garantit la positivité du jacobien des transformations associées.

L'inverse de τ_L n'est en général pas calculable explicitement. Par ailleurs, nous n'aurons à calculer l'antécédent d'un point physique par la transformation τ_L que dans certains cas particuliers. Pour cela, nous utiliserons une méthode de Newton.

Nous définissons alors sur L , une base d'approximation composée des fonctions $(\psi_i^L)_{i=0,\dots,(d+1)^3-1}$. Ces fonctions sont transportées sur l'élément physique à partir des fonctions de base définies sur l'élément de référence

$$\forall \hat{x} \in \hat{H}, \psi_i^L \circ \tau_L(\hat{x}) = \hat{\psi}_i(\hat{x}). \quad (4.26)$$

C'est-à-dire $\psi_i^L = \hat{\psi}_i \circ \tau_L^{-1}$. En général, les fonctions de base ainsi définies ne sont pas polynomiales dans l'espace physique¹. Par ailleurs, nous prolongeons par 0 ces fonctions en dehors de l'élément L

$$\forall x \notin L, \quad \psi_i^L(x) = 0. \quad (4.27)$$

Rappelons que puisque nous utilisons une interpolation nodale, à chaque instant t , l'approximation W est identifiée par ses valeurs sur les points d'intégration des mailles L . Notons W_i les composantes vectorielles de W dans la base d'approximation relative à L . W s'exprime en un point d'intégration

1. Mais les estimations d'interpolation (2.69) de la démonstration de convergence qui suit restent valables pour des maillages non dégénérés

$x_i = \tau_L(\hat{x}_i)$ de L par

$$\begin{aligned}
W(x_i, t) &= \sum_{j=0}^{(d+1)^3-1} W_j(t) \psi_j^L(x_i) \\
&= \sum_{j=0}^{(d+1)^3-1} W_j(t) \hat{\psi}_j(\hat{x}_i) \\
&= W_i(t).
\end{aligned} \tag{4.28}$$

Ainsi, la solution est entièrement définie par ses valeurs aux points d'intégration volumiques. Nous stockerons donc en mémoire ces valeurs.

Nous allons maintenant décrire la formulation discrète obtenue en utilisant l'espace d'approximation que nous avons construit. Plus précisément, nous examinerons les approximations des intégrales présentes dans la formulation GD (2.34 page 35) obtenues en appliquant les formules de quadrature (4.3 page 64) et (4.11 page 65).

4.2 Approximation des intégrales

En partant d'une formulation faible de notre problème de départ, nous avons alors établi une formulation semi-discrète. Nous allons maintenant décrire plus précisément le calcul des différentes intégrales apparaissant dans (2.51 page 41) lorsque nous discrétisons la solution dans l'espace d'approximation décrite dans le paragraphe précédent. Pour cela, nous estimerons ces intégrales au moyen des formules de quadratures (4.3 page 64) et (4.11 page 65). Nous verrons alors apparaître diverses simplifications des estimations des intégrales. Ces simplifications sont présentées par exemple dans [16].

Remarque 8. *Les formules de quadrature (4.3 page 64) et (4.11 page 65) reste exactes si la transformation géométrique τ_L est affine.*

Notons les décompositions des champs W et $\partial_t W$ dans la base d'approximation

$$W(x, t) = \sum_{L \in \mathcal{M}} \sum_{i=0}^{(d+1)^3-1} W_i^L(t) \psi_i^L(x) \text{ avec } W_i^L \in \mathbb{R}^6 \text{ et } x \in L \tag{4.29}$$

et

$$\partial_t W(x, t) = \sum_{L \in \mathcal{M}} \sum_{i=0}^{(d+1)^3-1} W'_i{}^L(t) \psi_i^L(x) \text{ avec } W'_i{}^L \in \mathbb{R}^6 \text{ et } x \in L. \quad (4.30)$$

Dans ce qui suit, nous nous intéresserons au calcul du terme (2.51 page 41) pour une fonction test ψ_i^L fixé.

4.2.1 Terme de masse

Le terme de « masse » est donné par l'intégrale

$$\begin{aligned} \int_L \partial_t W \psi_i^L dx &= \int_L \sum_{j=0}^{(d+1)^3-1} W'_j{}^L \psi_j^L \psi_i^L dx \\ &= \int_{\hat{H}} \sum_{j=0}^{(d+1)^3-1} W'_j{}^L \hat{\psi}_j \hat{\psi}_i \text{jac}(\tau_L) d\hat{x} \\ &= \delta_{i,j} \omega_i W'_i{}^L \text{jac}(\tau_L)(\hat{x}_i). \end{aligned} \quad (4.31)$$

Ainsi, la matrice de masse obtenue est diagonale et ne dépend que des poids d'intégration et du jacobien de la transformation géométrique τ_L sur les points d'intégration.

4.2.2 Terme de rigidité

Le terme de « rigidité » est donné par l'intégrale

$$\int_L F(W, W, \nabla \psi_i^L) dx = \int_{\hat{H}} F(W \circ \tau_L, W \circ \tau_L, \nabla \psi_i^L \circ \tau_L) \text{jac}(\tau_L) d\hat{x}. \quad (4.32)$$

Nous pouvons écrire le gradient $(\nabla \psi_i^L) \circ \tau_L$ sous la forme $(\tau'_L{}^T)^{-1} \hat{\nabla} \hat{\psi}_i$ où τ'_L est la matrice jacobienne de τ_L . Il est pratique d'introduire la comatrice $\text{com}(\tau'_L)$ de τ'_L . La comatrice est définie par

$$\text{com}(\tau'_L)^T \tau'_L = \det(\tau'_L) I. \quad (4.33)$$

Ainsi,

$$\int_L F(W, W, \nabla \psi_i^L) dx = \int_{\hat{H}} F(W \circ \tau_L, W \circ \tau_L, \text{com}(\tau'_L) \hat{\nabla} \hat{\psi}_i) d\hat{x}. \quad (4.34)$$

Ensuite, en appliquant la formule de quadrature (4.3 page 64)

$$\int_L F(W, W, \nabla \psi_i^L) dx \simeq \sum_{j=0}^{(d+1)^3-1} \omega_j F(W_j, W_j, \text{com}(\tau'_L) \hat{\nabla} \hat{\psi}_i(\hat{x}_j)). \quad (4.35)$$

Plusieurs termes de la somme précédente sont nuls, voir par exemple [16]. Nous pouvons les déterminer en appliquant les propriétés de fonctions de base. Cela nous permettra de diminuer significativement le volume de calcul. Par définition des fonctions de base, le gradient de la fonction de base $\hat{\nabla} \hat{\psi}_i$, avec $i = p(0) + (d+1)p(1) + (d+1)p(2)$, s'exprime

$$\hat{\nabla} \hat{\psi}_i(\hat{x}^0, \hat{x}^1, \hat{x}^2) = \begin{pmatrix} \ell'_{p(0)}(\hat{x}^0) \ell_{p(1)}(\hat{x}^1) \ell_{p(2)}(\hat{x}^2) \\ \ell_{p(0)}(\hat{x}^0) \ell'_{p(1)}(\hat{x}^1) \ell_{p(2)}(\hat{x}^2) \\ \ell_{p(0)}(\hat{x}^0) \ell_{p(1)}(\hat{x}^1) \ell'_{p(2)}(\hat{x}^2) \end{pmatrix}. \quad (4.36)$$

Examinons la valeur de la première composante de ce gradient sur un point d'intégration $\hat{x}_j = (\xi_{q(0)}, \xi_{q(1)}, \xi_{q(2)})$. Par définition des polynômes de Lagrange

$$\begin{aligned} \partial_0 \hat{\psi}_i(\hat{x}_j) &= \ell'_{p(0)}(\xi_{q(0)}) \ell_{p(1)}(\xi_{q(1)}) \ell_{p(2)}(\xi_{q(2)}) \\ &= \ell'_{p(0)}(\xi_{q(0)}) \delta_{p(1), q(1)} \delta_{p(2), q(2)}. \end{aligned} \quad (4.37)$$

Cette composante est donc nulle en tout point d'intégration dont les indices $q(1)$ et $q(2)$ ne correspondent pas aux indices $p(1)$ et $p(2)$ du point \hat{x}_i . Ces points sont les points alignés avec \hat{x}_i dans la direction $(1, 0, 0)^T$, c'est-à-dire les $d+1$ points de coordonnées

$$\hat{x}_f = (\xi_{q(0)}, \xi_{p(1)}, \xi_{p(2)}), \quad q(0) = 0 \dots d. \quad (4.38)$$

De la même manière, la seconde, respectivement la troisième, composante s'annule en tout point d'intégration, excepté les points alignés avec \hat{x}_i dans la direction $(0, 1, 0)$, respectivement la direction $(0, 0, 1)$.

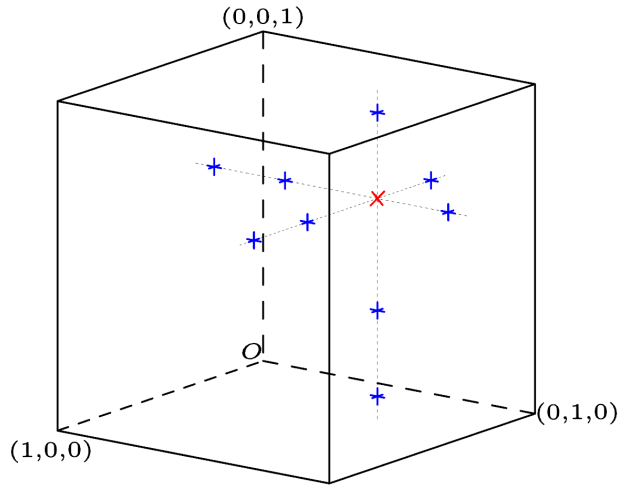


FIGURE 4.3 – Points d’interpolation alignés ($d = 3$). Pour calculer le gradient de la fonction de base associée au point « \times » rouge, seules les fonctions de base associées aux points « $+$ » bleus interviennent.

Afin de calculer l’intégrale (4.35 page précédente), nous n’avons à prendre en compte que les valeurs sur ces $3d+1$ points. De plus, en chacun de ces points, excepté \hat{x}_i , seule la composante du gradient correspondant à la direction d’alignement est non nulle. Cette propriété permet encore de simplifier la programmation des calculs de ce terme. Cette propriété pour $d = 3$ est illustrée sur la figure FIGURE 4.3 où le point d’indice i est représenté par une croix « \times » rouge et les points alignés par des croix « $+$ » bleues.

La matrice de rigidité est alors diagonale par bloc, chaque bloc correspondant à un élément du maillage. Le calcul de chacun de ses termes ne nécessite l’évaluation que de $3d+1$ termes et est local à chaque élément, *i.e.* qu’il n’y a pas d’interaction entre fonctions de base appartenant à des éléments différents. Cette propriété rend la parallélisation de ce terme aisée.

4.2.3 Terme de flux

Le terme de flux permet le couplage entre les mailles voisines. Notons W_L le champ W provenant de la maille L . Le champ W_R désignera le champ W provenant d’une maille R ayant une face en commun avec la maille L .

Notons également $(Q_f)_{f=0,\dots,5}$ les 6 faces de L image par τ_L des faces $(\hat{Q}_f)_{f=0,\dots,5}$ de \hat{H} .

Sur une face Q_f ne touchant pas le bord du domaine de calcul, en notant ds et $d\hat{s}$ les éléments de surface dans l'espace physique ou dans l'espace de référence, nous avons

$$\begin{aligned} & \int_{Q_f} F(W_L, W_R, n) \psi_i^L ds \\ &= \int_{\hat{Q}_f} F(W_L \circ \tau_L, W_R \circ \tau_L, \text{com}(\tau_L') \hat{n}) \psi_i^L \circ \tau_L d\hat{s} \end{aligned} \quad (4.39)$$

où $\text{com}(\tau_L')$ est la comatrice de τ_L' définie en (4.33 page 73).

Cette formule résulte de la définition de l'intégrale de surface (voir aussi la définition (4.11 page 65)). Nous approchons alors cette intégrale par la formule de quadrature (4.11 page 65). Notons $W_{f,i'}^L$ les champs au point d'intégration $x_{f,i'}$ d'indice i' de la face Q_f provenant des champs présents dans l'élément L . Plus précisément, ces champs valent

$$W_{f,i'}^L = \sum_{i=0}^{(d+1)^3-1} W_i^L \hat{\psi}_i(\hat{x}_{f,i'}). \quad (4.40)$$

Cette formule peut être simplifiée en tenant compte des propriétés des fonctions de base. Nous décrivons plus loin cette simplification.

Soit f_R l'indice de la face de l'élément R coïncidant avec la face f de L et j' l'indice du point d'intégration sur la face f_R de R correspondant au point $x_{f,i'}$, c'est-à-dire

$$\tau_R(\hat{x}_{f_R,j'}) = x_{f_R,j'} = \tau_L(\hat{x}_{f,i'}). \quad (4.41)$$

Notons alors $W_{f_R,j'}^R$ les champs provenant de l'élément R au point d'intégration d'indice j' sur la face Q_{f_R} de R . Ces champs peuvent être calculés par la formule

$$W_{f_R,j'}^R = \sum_{i=0}^{(d+1)^3-1} W_i^R \hat{\psi}_i(\hat{x}_{f_R,j'}). \quad (4.42)$$

Nous approchons alors l'intégrale précédente par

$$\begin{aligned} & \int_{Q_f} F(W_L, W_R, n) \psi_i^L ds \\ & \simeq \sum_{i'=0}^{(d+1)^2-1} \omega_{f,i'} \hat{\psi}_i(\hat{x}_{f,i'}) F(W_{f,i'}^L, W_{f_R,j'}^R, \text{com}(\tau_L' \hat{n})). \end{aligned} \quad (4.43)$$

De plus, la fonction de base $\hat{\psi}_i$ est non nulle en uniquement un point d'intégration de la face Q_f d'indice $\pi(f, i)$ d'après la propriété 11 page 68. Ce point est le projeté du point \hat{x}_i associé à la fonction de base $\hat{\psi}_i$.

La somme précédente se réduit donc à un seul terme

$$\begin{aligned} & \int_{Q_f} F(W_L, W_R, n) \psi_i^L ds \\ & \simeq \omega_{f,\pi(i)} \text{jac}(\tau_L)(\hat{x}_{f,\pi(f,i)}) \hat{\psi}_i(\hat{x}_{f,\pi(f,i)}) F(W_{f,\pi(i)}^L, W_{f_R,j'}^R, n \circ \tau_L(\hat{x}_{f,\pi(f,i)})). \end{aligned} \quad (4.44)$$

Afin de calculer ce flux, nous devons calculer les champs provenant des deux mailles voisines au point $x_{f,\pi(f,i)} = \tau_L(\hat{x}_{f,\pi(f,i)})$.

Puisque seules les fonctions de base associées aux points d'intégration se projetant sur $\hat{x}_{f,\pi(f,i)}$ sont non nulles en ce point, l'extrapolation du champ W_L n'utilise que les valeurs en ces points, soit $d + 1$ valeurs.

Dans le cas où les ordres d'approximation dans les deux mailles sont égaux, le point $\hat{x}_{f_R,j'} = \tau_R^{-1} \circ \tau_L(\hat{x}_{f,j})$ correspond à un point d'intégration sur une face de R , ainsi le calcul de W_R peut être effectué de la même manière que celui de W_L .

Dans le cas contraire, nous devons calculer le champ W_R au point $\hat{x}_{f_R,j'}$. En appliquant la définition de W_R

$$W_R(\tau_L(\hat{x}_{f_R,j'})) = \sum_{i=0}^{(d+1)^3-1} W_i^R \hat{\psi}_i(\hat{x}_{f_R,j'}). \quad (4.45)$$

Dans le cas d'une face touchant le bord du domaine, le flux numérique est remplacé par un flux de bord traduisant l'application d'une condition limite spécifique. L'évaluation du terme de flux nécessite comme précédemment le

calcul du champ W_L en un point d'intégration de cette face. Afin de garantir la stabilité du schéma, le flux de bord doit satisfaire la propriété 3 page 38 garantissant la décroissance de l'énergie du système.

Remarque 9. *Le calcul du terme décrit dans cette section fait intervenir des champs provenant de plusieurs éléments. Plus précisément, deux fonctions de base interagissent si elles appartiennent au même élément ou à des éléments partageant une face. Cela rendra la parallélisation du calcul de ce terme plus délicate et plus coûteuse en temps.*

Conclusion :

Dans les sections précédentes, nous avons défini un espace d'approximation formé de compositions entre une transformation géométrique et des produits tensoriels de polynômes de Lagrange. Ces polynômes de Lagrange sont construits à partir de points définissant une formule de quadrature les intégrant exactement. Nous avons également défini une formule de quadrature permettant d'estimer les intégrales surfaciques. En appliquant cette discrétisation et ces formules de quadrature, nous avons obtenu une estimation des intégrales présentes dans la formulation GD. L'utilisation de cette base et de ces formules de quadrature a permis différentes simplifications des calculs.

Toutefois, les formules de quadrature ne sont pas exactes lorsque les éléments de maillage sont trop déformés. Nous allons maintenant examiner la validité de ces approximations.

4.2.4 Validité de l'intégration numérique

Afin d'estimer les intégrales apparaissant dans la formulation GD, nous appliquons un changement de variables pour nous ramener à l'élément de référence puis utilisons les formules de quadrature (4.3 page 64) et (4.11 page 65).

La qualité de la formulation semi-discrète ainsi obtenue est liée d'une part à la précision de l'intégration numérique choisie et d'autre part à sa compatibilité avec la formule d'intégration par partie.

La compatibilité entre l'intégration numérique et l'intégration par parties assure, par exemple, la préservation des états constants, comme expliqué dans [42] et [43]. Si l'intégration numérique ne satisfait pas une propriété d'intégration par parties discrète, des ondes parasites peuvent apparaître, même lorsque l'approximation des champs est d'ordre élevé. Nous sommes conscients de ce défaut lié à l'intégration numérique, mais nous avons conservé,

pour des raisons de temps de calcul, dans nos calculs numériques les quadratures de Gauss-Legendre « naïves ». Pour être le plus complets possible, nous montrons tout de même dans ce paragraphe qu'il est possible en théorie de récupérer des propriétés d'intégration par parties discrètes. Les remarques qui suivent sont extraites de [42] et [43].

Examinons le comportement de la formule de Stokes lorsque nous l'appliquons à une fonction ψ avec intégration numérique comme effectué dans les sections 4.2.2 page 73 et 4.2.3 page 75.

$$\begin{aligned} 0 &= \int_L \nabla \psi - \int_{\partial L} \psi n \\ &= \int_{\hat{H}} \text{com}(\tau') \hat{\nabla} \hat{\psi} - \int_{\partial \hat{H}} \hat{\psi} \text{com}(\tau') \hat{n}. \end{aligned} \quad (4.46)$$

Ces quantités sont alors estimées en utilisant les formules de quadrature. Dans le cas où la transformation géométrique τ est affine, la matrice $\text{com}(\tau')$ est constante et les intégrales sont calculées de manière exacte. La formule d'intégration par partie est vérifiée de manière discrète.

Dans le cas d'une transformation non affine, l'intégrande n'est plus polynomial. La formule de quadrature devient inexacte et la formule d'intégration par partie n'est plus vérifiée.

Afin de retrouver une propriété d'intégration par partie discrète, Kopriva et Gassner proposent dans [43] de modifier l'intégration surfacique. Pour cela, définissons l'opérateur d'interpolation I qui associe à une fonction définie sur l'élément de référence son interpolé dans la base d'approximation

$$I(f) = \sum_{i=0}^{(d+1)^3-1} f(\hat{x}_i) \hat{\psi}_i. \quad (4.47)$$

Ainsi une formule de quadrature modifiée, appliquée au terme volumique peut être

$$\begin{aligned} \int_{\hat{H}} \text{com}(\tau') \hat{\nabla} \hat{\psi} &\simeq \int_{\hat{H}} I(\text{com}(\tau') \hat{\nabla} \hat{\psi}) \\ &= \int_{\hat{H}} I(I(\text{com}(\tau')) I(\hat{\nabla} \hat{\psi})). \end{aligned} \quad (4.48)$$

La formule d'intégration surfacique, quant à elle, requiert une estimation de

la valeur intégrée aux points d'intégration surfaciques. Dans la section 4.2.3 page 75, nous calculons ces valeurs en utilisant leurs définitions, c'est-à-dire que nous calculons la comatrice $\text{com}(\tau')$ et le gradient $\hat{\nabla}\hat{\psi}$ aux points d'intégration surfaciques. L'approche proposée par Kopriva consiste à remplacer ces valeurs exactes par les valeurs prises par l'interpolé $I(I(\text{com}(\tau'))I(\hat{\nabla}\hat{\psi}))$ aux points surfaciques.

$$\begin{aligned} \int_{\partial\hat{H}} \hat{\psi}\text{com}(\tau')\hat{n} &\simeq \int_{\partial\hat{H}} I(\hat{\psi}\text{com}(\tau'))\hat{n} \\ &= \int_{\partial\hat{H}} I(I(\hat{\psi})I(\text{com}(\tau')))\hat{n}. \end{aligned} \quad (4.49)$$

Ainsi puisque la formule de quadrature est exacte pour tout interpolé, la formule d'intégration par partie est vérifiée de façon discrète, même si l'intégration numérique est inexacte.

Il est également possible d'obtenir une propriété d'intégration par partie discrète en modifiant la base d'approximation. Par exemple, cette propriété est vérifiée en prenant les fonctions de base utilisées par Cohen, Ferrières et Pernet dans [16] définies par

$$\psi_i^p \circ \tau_L(\hat{x}) = \tau_L'^{T-1} \hat{\psi}_i(\hat{x}) e_p \quad (4.50)$$

où e_p est le $p^{\text{ième}}$ vecteur de base de la base canonique de \mathbb{R}^3 .

Même si cette approche permet d'assurer sur le plan discret une estimation d'énergie et la préservation des états constants, l'utilisation de cette intégration numérique n'est pas vraiment adaptée aux calculs parallèles sur GPU. En effet, celle-ci complexifie le calcul des termes de surface qui constituent la partie où le parallélisme est le moins efficace. De plus, les optimisations obtenues dans le cadre des équations de Maxwell ne sont pas transposables dans un cadre générique. Ces optimisations reposent sur la propriété suivante du rotationnel

$$\begin{aligned} \nabla \times \psi &= \left(\tau_L'^{T-1} \hat{\nabla} \right) \times \left(\tau_L'^T \hat{\psi} \right) \\ &= \text{jac}(\tau_L')^{-1} \tau_L' \hat{\nabla} \times \hat{\psi} \end{aligned} \quad (4.51)$$

qui n'est bien sûr pas adaptable à un système de Friedrichs arbitraire.

4.3 Schéma temporel Runge-Kutta 2

Nous avons donné une démonstration de convergence du schéma dans le cas semi-discrétisé en espace. Avant le passage sur ordinateur, il est indispensable de fournir également une technique de discrétisation en temps. Le choix du schéma en temps est crucial. Il faut en effet s'assurer de la précision et de la stabilité du schéma couplé temps-espace.

Nous avons précédemment décrit le calcul de la dérivée en temps $\partial_t W$ du champ en fonction du champ W à un instant donné t .

Dans cette section, nous noterons \mathcal{G} , l'application qui à un champ $W \in \mathcal{V}_h$ et un instant t associe sa dérivée en temps obtenue par la méthode GD décrite précédemment.

$$\partial_t W = \mathcal{G}(W, t) \tag{4.52}$$

Soit Δt un réel strictement positif. Notons W_n la solution numérique de (2.51 page 41) au temps $n\Delta t$.

Supposons connu W_n pour un entier n donné.

En pratique, nous calculons le plus souvent W_{n+1} par la méthode de Runge-Kutta d'ordre 2.

$$W_{n+1/2} = W_n + \frac{\Delta t}{2} \mathcal{G}(W_n, t) \tag{4.53}$$

$$W_{n+1} = W_n + \Delta t \mathcal{G}(W_{n+1/2}, t + \frac{\Delta t}{2}) \tag{4.54}$$

4.3.1 Stabilité du schéma temporel

Nous allons maintenant étudier la stabilité du schéma GD couplé à une intégration en temps de Runge-Kutta 2. Pour cela, nous suivrons l'approche décrite dans les thèses de Jund [39] et Crestetto [19].

Considérons le problème aux limites (2.3 page 28) avec des conditions aux limites satisfaisant les conditions d'existence et d'unicité de la solution. Soit W la solution approchée de ce système obtenue en appliquant les schémas GD et de Runge-Kutta 2.

Notons alors A la matrice liant la dérivée en temps $\partial_t W$ au champ W . Le schéma numérique s'écrit en simplifiant

$$\partial_t W = AW. \quad (4.55)$$

Dans ce cas, la solution W peut être exprimée par

$$W(t) = \exp(At)W(0). \quad (4.56)$$

Dans ce cas particulier, tous les schémas d'intégration en temps à un pas sont équivalents, une fois que l'ordre du schéma est fixé. Soit Δt le pas de temps et $t_n = n\Delta t$. Un développement de Taylor donne l'approximation à l'ordre p

$$W(t_{n+1}) = \left(\sum_{i=0}^p \frac{\Delta t^i}{i!} A^i \right) W(t_n). \quad (4.57)$$

Notons

$$\tilde{A} = \sum_{i=0}^p \frac{\Delta t^i}{i!} A^i. \quad (4.58)$$

Le schéma est stable si l'inégalité suivante est vraie

$$\|W(t_{n+1})\| \leq \|W(t_n)\|. \quad (4.59)$$

D'après (4.57), ceci est vrai si la norme de \tilde{A} est inférieure ou égale à 1. En particulier, le schéma est stable si le rayon spectral de \tilde{A} est inférieur ou égale à 1.

Les valeurs propres de \tilde{A} sont de la forme $\sum_{i=0}^p \frac{(\Delta t \lambda)^i}{i!}$ où λ est une valeur propre de A .

Le schéma d'ordre p est donc stable pour un pas de temps Δt si toutes les valeurs propres λ de A sont telles que $\Delta t \lambda$ appartiennent à

$$S_p = \left\{ \mu \in \mathbb{C}, \left| \sum_{i=0}^p \frac{\mu^i}{i!} \right| \leq 1 \right\}. \quad (4.60)$$

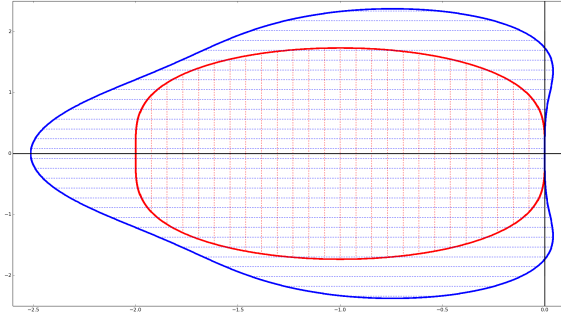


FIGURE 4.4 – Zone de stabilité dans le plan complexe pour un schéma d’ordre 2 (hachurée verticalement en rouge) et pour un schéma d’ordre 3 (hachurée horizontalement en bleu). Les axes réel et imaginaire sont représentés par les lignes pleines noires.

Dans notre cas, la zone de stabilité est

$$S_2 = \left\{ \mu \in \mathbb{C}, \left| 1 + \mu + \frac{\mu^2}{2} \right| \leq 1 \right\}. \quad (4.61)$$

En particulier, les valeurs propres de A doivent avoir des parties réelles négatives².

La FIGURE 4.4 représente la zone de stabilité des schémas RK2 (hachurée verticalement en rouge) et RK3 (hachurée horizontalement en bleu) dans le plan complexe. Nous pouvons remarquer sur cette figure que la zone de stabilité du schéma RK3 englobe la zone de stabilité du schéma RK2 et une partie de l’axe imaginaire (alors que celle du schéma RK2 lui est tangente).

Nous pouvons alors vérifier numériquement la stabilité du schéma en calculant les valeurs propres λ de la matrice A puis en vérifiant l’existence d’un réel strictement positif Δt tel que les $\Delta t \lambda$ appartiennent tous à la zone de stabilité S_2 .

Le calcul de la matrice A nécessite le calcul par le schéma GD de la dérivée en temps associée à autant de champs W que le nombre de degrés de liberté. Pour un maillage contenant N hexaèdres et une approximation à l’ordre d , cela représente $6N(d+1)^3$ calculs de dérivées en temps. Ce calcul est coûteux

2. En toute rigueur, le rayon spectral d’une matrice n’est pas toujours le plus grand module de ces valeurs propres, mais c’est le cas si la matrice est diagonalisable.

en temps pour un maillage contenant un grand nombre d'éléments.

Nous pouvons néanmoins estimer la stabilité dans un élément donné. Pour cela, nous ne considérons que la partie volumique du schéma GD qui est local à l'élément et remplaçons les termes de flux faisant intervenir les champs dans les éléments voisins par un flux représentant une condition aux limites. Nous donnons un exemple de cette procédure dans la section 4.3.3 où nous observerons l'influence de la dissipation du flux de bord.

4.3.2 Une condition de stabilité *a priori*

Ce qui précède permet de vérifier numériquement la stabilité de la méthode sur un maillage donnée et peut servir d'outil de diagnostic. Néanmoins, il est nécessaire d'avoir une estimation *a priori* du pas de temps maximal assurant la stabilité du schéma. La définition d'un tel pas de temps par un critère de type CFL a fait l'objet de différents travaux, voir par exemple [15, 45] ou encore [16, 24] pour une estimation de la stabilité du schéma GD associé à une intégration en temps de type saute-mouton. Nous utiliserons alors la condition de stabilité décrite dans [45] sous la forme

$$\Delta t \leq \frac{\Delta x}{c(2d+1)}. \quad (4.62)$$

Toutefois, Rider et Lowrie ont démontré dans [62] que le couplage du schéma GD et d'une approximation en temps de type RK peut conduire à un schéma inconditionnellement instable en particulier en utilisant une approximation spatiale d'ordre 2. Ce résultat peut être obtenu en analysant la répartition des valeurs propres de la matrice A vérifiant (4.55 page 82) dans le cas unidimensionnel sur un maillage uniforme et infini. Dans ce cas, les valeurs propres de cette matrice sont incluses dans le demi-plan complexe $(\mathbb{R}_+^* \times i\mathbb{R}) \cup 0$ et forment au voisinage de zéro une courbe tangente à un ordre élevé à l'axe imaginaire pur. Malheureusement, le contour de la zone de stabilité de la méthode RK2 est quant à lui tangent à un ordre inférieur, y rendant impossible l'inclusion des valeurs propres de A .

Toutefois lorsque le maillage est fini, les valeurs propres de A forme un sous-ensemble discret de \mathbb{C} et peuvent donc être ramenée dans la zone de stabilité suivant une condition CLF plus contraignante.

Il est également possible de retrouver une condition de stabilité plus faible est utilisant un flux numérique de type Lax-Friedrichs modifié (voir [62]) ou encore en utilisant un schéma RK3 dont la zone de stabilité contient

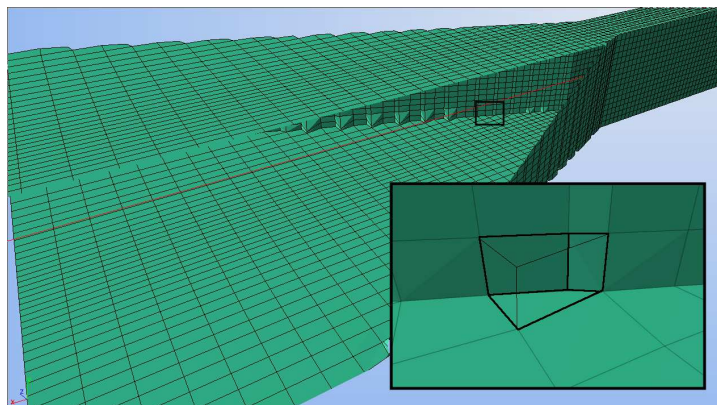


FIGURE 4.5 – Partie surfacique d’un maillage (tronquée par un plan de coupe) contenant des éléments fortement déformés. Le zoom met en évidence un hexaèdre déformé jusqu’à devenir un prisme.

une partie de l’axe imaginaire pur centré en zéro visible sur la figure FIGURE 4.4 page 83.

Malgré cela, nous n’avons pas constaté de divergence de la méthode lors des simulations effectuées, et ce même à l’ordre 2.

4.3.3 Une application du diagnostic numérique de stabilité

Dans ce qui précède, nous avons décrit une technique numérique permettant de diagnostiquer une instabilité. Appliquons alors cette technique aux deux représentations de la condition aux limites de conducteur parfait introduites dans la section 3.1 page 49.

La figure FIGURE 4.5 représente une structure surfacique d’un maillage. Le maillage est généré en réalisant un maillage cartésien de la structure puis en déformant les éléments pour rendre le maillage conforme à la structure simulée. Ce processus produit des éléments ayant une forte déformation dans les zones où est présente une structure oblique. Le zoom permet de mettre en évidence un tel élément. Sur trois des faces de cet élément, un flux représentant une condition aux limites métal est appliqué et sur les autres faces un flux décentré est appliqué.

Nous allons donc étudier la stabilité du schéma dans un tel élément. Consi-

dérons donc un élément dont les coordonnées des nœuds sont

$$\begin{aligned}
S_0 &= (0, 0, 0), \\
S_1 &= (1, 0, 0), \\
S_2 &= (1, 1, 0), \\
S_3 &= (0, 1, 0), \\
S_4 &= (-1, 0, 0), \\
S_5 &= (0, -0, 1), \\
S_6 &= (0, 1, 1), \\
S_7 &= (-1, 1, 0).
\end{aligned} \tag{4.63}$$

Nous appliquons sur les faces 0, 3 et 4 (voir (4.4 page 64)) un flux de type métal et sur les autres faces un flux représentant une condition aux limites de Silver-Müller définie dans la section 3.4 page 55. Ce flux est défini par

$$F_{SM}(\mathcal{W}, n) = F(\mathcal{W}, 0, n) = \mathcal{A}^i n_i^+ \mathcal{W}. \tag{4.64}$$

Ainsi, il aura la même influence sur la stabilité du schéma que le flux décentré. Cette configuration est représentée sur la figure FIGURE 4.6 page suivante sur laquelle les faces métalliques sont représentées en vert. Pour les flux de type métal dissipatif (3.8 page 51) et non dissipatif (3.7 page 51), nous calculons la matrice A liant les champs électromagnétiques à leur dérivée en temps (voir (4.57 page 82)).

La figure FIGURE 4.7 page suivante représente les valeurs propres obtenues en utilisant les flux de bord dissipatif (3.8 page 51) et non dissipatif (3.7 page 51). Dans les deux cas les matrices A ont été modulées par un pas de temps devant assurer la stabilité du schéma, dans notre cas

$$\Delta t = \frac{3 \times 10^{-2}}{c}. \tag{4.65}$$

Dans le cas de l'utilisation d'un flux non dissipatif, nous voyons apparaitre des valeurs propres en dehors de la zone de stabilité du schéma. Nous pourrions toutefois retrouver un schéma stable en diminuant le pas de temps, dans le cas présent en le divisant pas trois, entraînant une augmentation inversement proportionnelle du temps de calcul. L'utilisation du flux dissipatif nous permet de conserver la stabilité du schéma, sans contraindre le pas de temps.

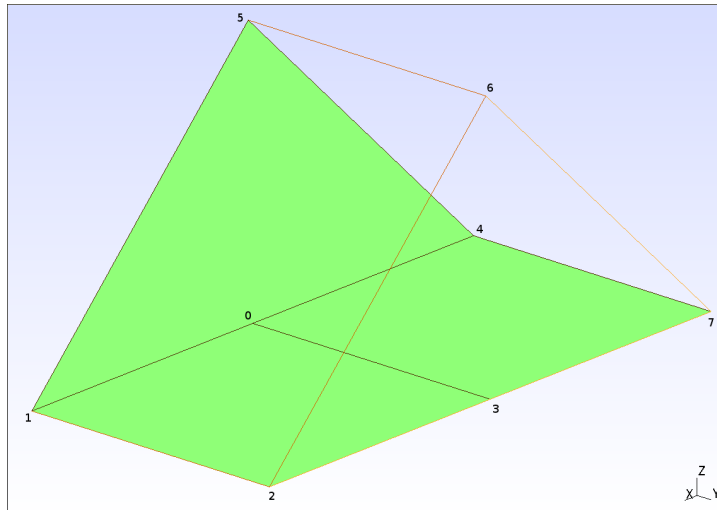


FIGURE 4.6 – Hexaèdre déformé dans lequel nous étudions la stabilité du schéma GD. Les faces vertes portent un flux de type métal, les autres faces portent un flux de type Silver-Müller.

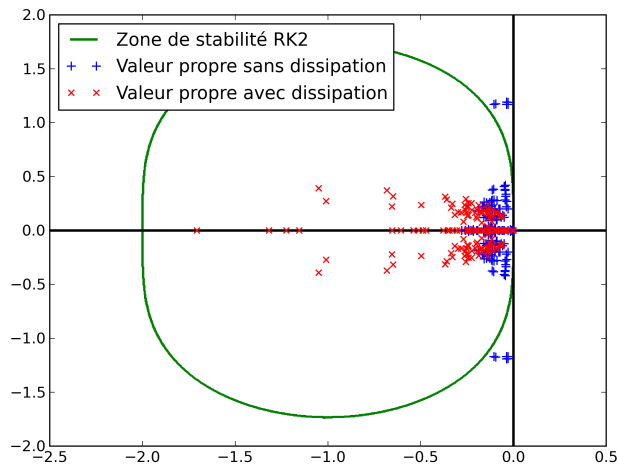


FIGURE 4.7 – Valeurs propres dans le plan complexe pour les conditions aux limites dissipative (\times rouge, (3.8 page 51)) et non dissipative ($+$ bleu (3.7 page 51)). Le pas de temps est fixé à une valeur qui assure la stabilité du schéma discret. En vert, nous avons représenté le contour de la zone de stabilité du schéma. Nous constatons que pour « faire rentrer » les valeurs propres bleues dans la zone verte, il faudrait diminuer fortement le pas de temps.

Conclusion

En conclusion, dans ce chapitre, nous avons passé en revue diverses optimisations et simplifications possibles de la méthode GD lorsque le maillage est constitué d'hexaèdres déformés. Ces optimisations sont essentiellement de deux types :

- dans le calcul des intégrales de volumes (terme de rigidité), le gradient de la fonction de base i s'annule sur tous les points de Gauss qui ne sont pas alignés avec le point de Gauss i ;
- dans le calcul des termes de flux, la reconstruction des champs en un point de Gauss de la face ne fait intervenir que les fonctions de base associées à des points de Gauss volumiques alignés avec ce point.

Ces optimisations et simplifications accélèrent déjà de façon très importante la programmation de la méthode sur un ordinateur séquentiel. Notre but est d'accélérer encore les calculs au moyen d'une parallélisation de la méthode sur un ou plusieurs GPU.

Nous avons également présenté le schéma temporel RK2 utilisé pour l'intégration en temps. Nous avons donné une condition de type CFL permettant de calculer un pas de temps pour lequel le schéma ne divergera pas, ainsi qu'un moyen pratique permettant de s'assurer de la stabilité du schéma.

Nous allons maintenant mettre en place l'algorithme parallèle multi-GPU utilisant la méthode GD que nous avons présentée. Cet algorithme est appliqué aux équations de Maxwell (1.1 page 8) et au modèle de PML (1.50 page 24).

Chapitre 5

Programmation sur GPU

Dans ce chapitre, nous décrivons l'implémentation sur un ou plusieurs GPU de la méthode GD sur hexaèdres décrite au chapitre précédent. Nous allons exploiter la puissance des cartes graphiques récentes qui permettent de réaliser de façon massivement parallèle des calculs généralistes. Ces « General Purpose Graphic Processing Units » (GPGPU ou plus simplement GPU) possèdent une puissance crête élevée, de l'ordre du Téraflops. Ce type de matériel est de plus en plus utilisé dans la simulation numérique. Nous trouvons cette utilisation des GPU dans la résolution des équations de Maxwell [10, 40], de Vlasov-Maxwell [20], la simulation de gaz et fluides [21, 32], et bien d'autres domaines.

Nous avons réalisé cette programmation en C++ en utilisant l'environnement OpenCL (« Open Computing Language » [29]) pour piloter le parallélisme à l'intérieur des unités de calcul d'un GPU. Cette bibliothèque permet d'exploiter la puissance de calcul de divers types d'accélérateurs (GPU, CPU multi-cœurs, coprocesseurs, *etc.*) produit par divers constructeurs (NVidia, AMD, Intel) sous différents systèmes d'exploitation (Linux, Windows, Mac). Ainsi l'effort de programmation non négligeable fourni pour paralléliser les calculs sur GPU est aussi exploitable sur un processeur multicœur plus classique.

Les GPU actuels ont la particularité de posséder un volume de mémoire limité, ne dépassant pas quelques gigaoctets. Il est donc important, à la fois pour pouvoir traiter des volumes de données importants et pour atteindre une puissance encore plus élevée, de pouvoir exploiter en même temps plusieurs GPU. Ainsi, nous avons également parallélisé les calculs sur plusieurs périphériques en utilisant la bibliothèque MPI (Message Passing Interface). Celle-ci offre la possibilité de lancer plusieurs processus indépendants qui ef-

fectuent les calculs sur différentes parties du maillage et communiquent par un système de messages.

Dans un premier temps, nous présenterons la parallélisation OpenCL des calculs, puis nous décrirons une première stratégie synchrone de parallélisation MPI. Nous verrons au chapitre suivant comment améliorer cette approche au moyen d'une stratégie asynchrone.

5.1 Parallélisation OpenCL

Un premier niveau de parallélisation consiste donc en l'utilisation de la bibliothèque OpenCL. Dans cette section, nous présentons la stratégie de parallélisation utilisée. Dans un premier temps, nous abordons la bibliothèque OpenCL et les spécificités de parallélisation sur GPU. Ensuite, nous décrivons une implémentation pouvant être utilisée sur différents types de maillages. Enfin, nous utilisons les spécificités de l'espace d'approximation sur maillage hexaédrique afin d'optimiser la parallélisation pour ce type spécifique de maillage.

5.1.1 La bibliothèque OpenCL

OpenCL est une interface de programmation (« Application Programming Interface » ou API) maintenue par le Khronos Group [29], qui gère aussi le standard graphique OpenGL. Cette API a été d'abord inventée par Apple, puis confiée à Khronos en 2008, dans l'espoir qu'elle devienne un standard. La première version est sortie en 2008. Aujourd'hui, la version courante est la version 1.2. Différents constructeurs (AMD, Nvidia, Intel, ...) ont développé des pilotes permettant d'utiliser OpenCL pour paralléliser des calculs sur leurs matériels. Les pilotes OpenCL sont disponibles pour la plupart des accélérateurs : CPU multicoeur des marques AMD, Intel, ARM, IBM ; GPU des marques AMD, NVIDIA, ARM MALI ; ou coprocesseur Intel Xeon Phi ; *etc.*

Cette bibliothèque est donc relativement portable. Elle peut être utilisée dans un environnement Windows, Linux ou Mac.

Périphérique OpenCL

Dans l'abstraction OpenCL, l'hôte (« host ») est le processeur qui exécute les instructions du programme principal, assure la soumission des calculs sur les périphériques de calculs ainsi que les tâches annexes (pré- et post-traitement des données, *etc.*). Les périphériques de calculs (« devices ») ou accélérateurs exécuteront les noyaux de calculs intensifs. Ces calculs sont programmés en utilisant un langage proche du C appelé OpenCL-C. Un tel programme est appelé *kernel*. Les sources des *kernels* peuvent être construites et compilées lors de l'exécution du programme hôte, nous permettant ainsi d'y inscrire « en dur » des informations contextuelles.

La soumission des tâches au périphérique est effectuée via une file d'attente. Les tâches peuvent être des transferts de données entre l'hôte et le périphérique OpenCL, ou des exécutions de *kernels*. L'exécution des tâches peut être gérée de plusieurs manières. Les tâches peuvent être soumises de manière synchrone. Elles sont alors effectuées dans l'ordre de soumission et l'exécution du programme hôte est stoppée durant l'exécution de la tâche.

L'exécution des tâches peut être également effectuée de manière asynchrone. Dans ce cas, chaque tâche est associée à un évènement OpenCL qui permet de connaître le statut d'exécution de la tâche : en attente, soumise, en cours d'exécution ou terminée. L'exécution du programme hôte continue durant l'exécution des tâches. Afin de garantir le bon déroulement des calculs, l'exécution d'une tâche peut être conditionnée par la terminaison d'un jeu de tâches à l'aide d'une liste d'évènements. Il est classique de représenter les dépendances entre les évènements au moyen d'un graphe de tâches. Les figures FIGURE 7.2 page 141 et FIGURE 7.3 page 142 représentent de tels graphes.

Du point de vue de l'exécution d'un *kernel*, le périphérique est vu sous la forme d'un ensemble d'unités de travail, les *work-items*, rassemblés en groupes de travail, les *work-groups*, ainsi que de plusieurs espaces de mémoire. La figure FIGURE 5.1 page suivante représente un périphérique (« device ») OpenCL. Les nombres de *work-items* et de *work-groups* associés à l'exécution d'un *kernel* sont spécifiés lors de la soumission de la tâche dans la file d'attente. Tous les *work-items* exécutent alors le même programme. Comme les *work-items* d'un même *work-group* partagent la même unité d'instruction¹ les instructions conditionnelles conduisent à l'exécution de toutes les branches. Seules les écritures sont désactivées dans les branches rejetées. Ainsi, nous devons limiter au maximum le nombre de branches présentes dans les *kernels*.

1. C'est le principe du parallélisme SIMD : Single Instruction Multiple Data.

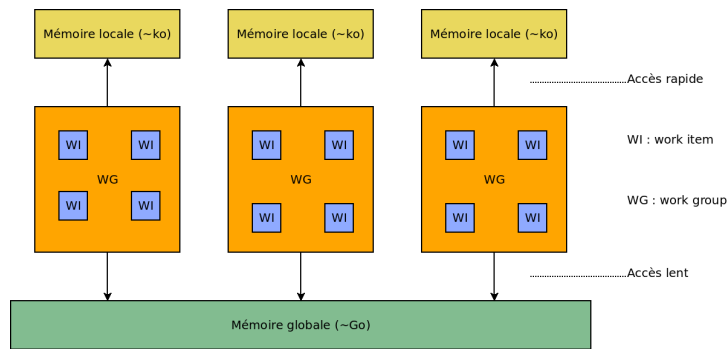


FIGURE 5.1 – Schématisation d'un périphérique OpenCL comprenant trois *work-groups* formés de 4 *work-items* chacun.

Lors de l'exécution d'un *kernel* par le périphérique OpenCL, nous avons accès à des données contextuelles qui permettent de piloter la parallélisation. Nous pouvons par exemple connaître le numéro du *work-item* et le numéro du *work-group*. Ces données permettent alors de paralléliser les calculs de la manière souhaitée.

La mémoire du périphérique est constituée de différentes parties. D'abord une mémoire constante (identifiée par le préfixe `__constant`). Cette mémoire est d'accès rapide et accessible par tous les *work-items*. Les données dans cet espace sont allouées et définies lors de la compilation des *kernels* et leurs valeurs ne peuvent pas être changées lors de l'exécution. Ensuite, chaque *work-group* a accès à une mémoire cache locale (identifiée par le préfixe `__local`). Les accès à cette espace de mémoire sont très rapides, mais les valeurs qui y sont stockées ne sont accessibles que par les *work-items* d'un *work-group* et sont perdues à la fin de l'exécution d'un *kernel*. En outre, cette mémoire est petite, de l'ordre de quelques dizaines de kilo-octets, et rapide d'accès. Nous stockerons ici des données temporaires partagées au sein d'un *work-group* nécessitant des accès répétés. Enfin, le périphérique contient une mémoire globale (identifiée par le préfixe `__global`). Celle-ci est de l'ordre de quelques giga-octets dans le cas d'une carte graphique. Cette mémoire est partagée par tous les *work-items* exécutant un *kernel*. Les lectures et écritures y sont relativement lentes. L'efficacité des accès à cette mémoire globale est alors conditionnée par leur ordonnancement. La façon la plus efficace de procéder est d'accéder de manière « coalescente » : des *work-items* dont les numéros se suivent doivent accéder à des données qui sont rangées de manière contiguë.

Dans les présents travaux, nous avons plus spécifiquement adapté l'implémentation de l'algorithme GD aux périphériques de type carte graphique

(GPU). Ce type de périphérique possède un faible volume de mémoire. Nous avons alors choisi de calculer quand nécessaire certains paramètres constants afin d'épargner un stockage. De plus, il est nécessaire d'utiliser les ressources de mémoire cache locale pour stocker les résultats intermédiaires, mais aussi certaines données dont l'accès est fréquent.

5.1.2 Zone de maillage homogène

Afin d'améliorer la parallélisation des calculs. Les mailles sont réparties dans des zones homogènes. Chaque zone contient un unique type de maille ainsi qu'une même base d'approximation. Les différentes phases du calcul de la dérivée temporelle des champs sont effectuées zone par zone.

Lors de la création de la représentation du maillage, les éléments sont triés suivant leur nature. Cela nous permet d'une part de générer les zones homogènes. D'autre part, ce tri assure que les données relatives aux éléments d'une zone soient stockées de manière contiguë en mémoire. Ainsi, une grande partie des lectures et écritures pourront être effectuées de manière coalescente garantissant leur efficacité. Ce sera le cas notamment pour les calculs des termes volumiques de la méthode GD.

Toutefois, le calcul des termes surfaciques nécessite l'accès aux champs présents dans plusieurs mailles qui sont potentiellement dans plusieurs zones. Cette phase de calcul ne peut pas bénéficier de toutes les optimisations pouvant être mises en place dans le cas d'une unique zone.

Pour une zone homogène, nous avons programmé divers types d'algorithmes. Ces algorithmes sont plus ou moins optimisés suivant les hypothèses retenues dans la zone. Par exemple, nous avons programmé un algorithme générique qui pourrait fonctionner sur des bases nodales pour des géométries quelconques (tétraèdres ou prismes par exemple). Nous avons aussi implémenté des algorithmes optimisés pour les maillages d'hexaèdres.

Disposer de plusieurs implémentations permet de vérifier l'efficacité des optimisations.

5.1.3 Implémentation générique

Nous décrivons dans cette section, une implémentation parallèle générique de l'algorithme GD utilisant OpenCL. La généricité de cette implémentation signifie qu'elle peut s'appliquer à différents types de maillages, différents types

de quadrature et différents systèmes de lois de conservation. Néanmoins, la base d'approximation dans une maille donnée est toujours une base nodale, ou plus précisément, une base constituée de polynômes de Lagrange associés aux points d'intégration dans la maille volumique.

Nous supposons avoir accès à des fonctions et variables décrivant d'une part les fonctions de base et d'autre part la géométrie et l'interpolation. Ces dernières sont dépendantes du type d'éléments pris en compte.

Supposons connus les W_n au temps $t_n = n\Delta_t$. Décrivons alors, les étapes permettant de calculer les champs W_{n+1} au pas de temps suivant. Le calcul est effectué de la manière suivante

- Calcul $\partial_t W$ de la dérivée en temps au temps t_n .
- Estimation du champ $W^* = W_n + \Delta t/2 \partial_t W$ en $t_{n+1/2}$.
- Calcul $\partial_t W$ de la dérivée en temps à partir du champ W^* .
- Avancée d'un pas de temps $W_{n+1} = W_n + \Delta t \partial_t W$.

Les première et troisième étapes sont les calculs de dérivée en temps. Dans ces étapes, nous utilisons la méthode GD parallèle (2.51 page 41) pour calculer la dérivée en temps à partir des champs donnés sur le maillage (W_n ou W^*).

Nous partageons ce calcul en trois sous-étapes

- Calcul des termes volumiques décrit en 4.2.2.
- Calcul des termes surfaciques décrit en 4.2.3.
- Terme de masse décrit en 4.2.1.

Chacune des étapes est traitée par un *kernel* OpenCL.

Étape de type Euler

Les deuxième et quatrième étapes consistent en l'avancée d'un pas de temps, similaire à une méthode d'Euler. La parallélisation de cette étape est aisée. Nous associons un *work-item* à chaque composante des champs dans la base d'approximation. Chacun effectue alors le calcul pour la composante qui lui est associée. Les lectures et écritures en mémoire sont dans ce cas parfaitement coalescentes et il n'est pas nécessaire d'utiliser les ressources de mémoire locale. Le code source de ce *kernel* est donné en 5.1. De plus, ce *kernel* n'utilise aucune donnée géométrique ou d'interpolation, il peut être ainsi appliqué à tout le maillage sans tenir compte de la répartition en zones.

```

__kernel void euler(__global const real* w,
                   const real dt,
                   __global const real* dtw,
                   __global real* wp1){

    // Numero du workitem
    int i = get_global_id(0);

    wp1[i] = w[i] + dt * dtw[i];

}

```

Algorithme 5.1 – Kernel de calcul de l'étape Euler.

Calcul des termes volumiques

Le *kernel* que nous décrivons ici calcule l'intégrale (4.35 page 74). Les calculs sont parallélisés sur un nombre de *work-items* égal au nombre de fonctions de base. Les *work-items* sont regroupés en *work-groups* correspondant chacun à un élément. Ils sont donc constitués d'un nombre de *work-items* égal au nombre de fonctions de base à support dans un élément. Nous noterons par la suite N le numéro du *work-group* courant, J le numéro du *work-item* courant et j le numéro du *work-item* dans son *work-group*.

Dans un premier temps, chaque *work-item* lit en mémoire globale les coordonnées des nœuds de l'élément auquel il est associé. Cette étape est nécessaire et requiert plusieurs lectures en mémoire non coalescentes, car le maillage est généralement non structuré.

Ensuite, les champs contenus dans l'élément N sont lus et stockés en mémoire locale. Cette lecture est faite de manière parallèle. Le *work-item* J lit la $J^{\text{ème}}$ composante du vecteur contenant les champs. Nous assurons la coalescence des lectures en choisissant l'ordre de rangement des valeurs des champs.

Nous associons alors un *work-item* à une fonction de base ψ_j^N . Nous calculons l'intégrale (4.35 page 74) en ajoutant les contributions de chaque point d'interpolation p de l'élément N . Pour cela, nous cherchons les coordonnées dans l'élément de référence ainsi que la comatrice de la jacobienne de la transformation géométrique τ associée à cet élément. Ces calculs utilisent des fonctions qui dépendent du type de l'élément de maillage de la zone.

Une signature OpenCL-C est imposée à chacune de ces fonctions.

Remarque 10. *Précisons que la définition effective de ces fonctions est ajoutée au code source des kernels lors de l'initialisation. De même, les données constitutives de zones, telles que le degré d'approximation, les nombres de noeuds et faces par élément, sont écrites sous forme d'un tableau stocké en mémoire constante dans le code source des kernels. Le code source des kernels est ensuite compilé par des appels à la bibliothèque OpenCL lors de la phase d'initialisation.*

Ensuite, nous lisons en mémoire locale le champ en ce point. Chaque *work-item* va donc lire tous les champs associés à l'élément N . Il est donc intéressant de stocker en mémoire locale ces champs. Nous ne faisons que M lectures en mémoire globale au lieu de M^2 , M étant le nombre de fonctions de base dans l'élément. Ces données permettent alors de calculer le terme (voir (4.35 page 74))

$$\omega_j F(W_j, W_j, \text{com}(\tau'_L) \hat{\nabla} \hat{\psi}_i(\hat{x}_j)) \quad (5.1)$$

à partir du flux numérique. Cette contribution est alors ajoutée à un vecteur temporaire, stocké en mémoire cache.

Enfin, nous ajoutons les valeurs temporaires au tableau en mémoire globale contenant les valeurs des dérivées en temps.

Calcul des termes surfaciques

Dans cette section, nous décrivons la parallélisation du calcul de l'intégrale surfacique (4.39 page 76). Nous supposons que les différentes mailles entrantes en jeu font toutes parties d'une même zone.

Dans ce *kernel*, nous ne calculons donc pas les intégrales sur les faces correspondantes à une interface entre zones. Pour l'instant, rappelons que nous souhaitons que le *kernel* s'applique à différents types de géométries. Ainsi, les optimisations permettant d'obtenir la formulation (4.44 page 77) sur les

hexaèdres ne s'appliquent pas ici. Nous utilisons la formulation (4.43 page 77)

$$\begin{aligned} & \int_{Q_f} F(W_L, W_R, n) \psi_i^L ds \\ & \simeq \sum_{i'=0}^{(d+1)^2-1} \omega_{f,i'} \hat{\psi}_i(\hat{x}_{f,i'}) F(W_{f,i'}^L, W_{f_R,j'}^R, \text{com}(\tau_L') \hat{n}) \end{aligned} \quad (5.2)$$

avec les notations du Chapitre 1.

Ces calculs sont effectués avec un nombre de *work-groups* égal au nombre d'éléments dans la zone. Chaque *work-group* est composé d'un nombre de *work-items* égal au maximum du nombre de points d'intégration présents sur les faces d'un élément et du nombre de points d'intégration volumiques. Prenons l'exemple d'une zone contenant 1000 éléments munie d'une approximation d'ordre 2. Chaque élément possède $(2+1)^3 = 27$ points d'intégration volumiques et $6 \times (2+1)^2 = 54$ points d'intégration surfaciques. Ainsi, le *kernel* sera exécuté par 1000 workgroups composés chacun de 54 *work-items*.

Nous noterons L le numéro du *work-groups* et de l'élément associé, J le numéro du *work-item* courant et j le numéro du *work-item* dans son *work-group*.

Dans un premier temps, nous stockons en mémoire locale les champs contenus dans l'élément L . Cette lecture est effectuée comme dans le calcul de l'intégrale volumique.

Les calculs sont alors effectués en deux étapes. La première étape est constituée du calcul des termes de la somme (5.2) et la seconde étape consiste en l'accumulation de ces termes. Ces deux étapes sont réalisées avec des répartitions des *work-items* différentes, d'abord sur les points de Gauss des faces, puis sur les points de Gauss des volumes.

Remarque 11. *Comme les nombres de points sur les faces et dans le volume sont différents, dans une des deux étapes certains work-items n'effectueront aucune opération. Jusqu'au degré 4, le nombre de points surfacique est plus important.*

Remarque 12. *Il serait également possible de calculer les flux sur chaque face et de les stocker en mémoire globale. Puis dans un second kernel, de lire ces flux et en ajouter les contributions à chaque élément. Cette approche n'a pas été retenue pour deux raisons. D'abord, elle requiert en volume de stockage supplémentaire alors que nous souhaitons économiser la mémoire. Ensuite, cette approche entraîne des accès mémoire non coalescents lors de ces deux étapes.*

Nous commençons par répartir les *work-items* sur les points d'intégration des faces. Pour cela, nous calculons un indice de face $0 \leq f \leq N_{face} - 1$ et un indice de point $0 \leq i' \leq M_{face} - 1$ à partir de l'indice j du *workitem* de manière à avoir

$$j = i' + M_{face}f \quad (5.3)$$

en notant N_{face} le nombre de faces des éléments et M_{face} le nombre de points d'intégration par face qui sont deux entiers constants dans chaque zone.

Chaque *work-item* calcule alors les valeurs des champs sur le point qui lui est associé. Pour cela nous calculons les coordonnées dans l'élément de référence de ce point, puis nous ajoutons toutes les contributions des fonctions de base en ce point. L'extrapolation des champs en ce point ne nécessite alors que des lectures en mémoire locale.

Nous calculons ensuite le champ provenant de la maille voisine R . Pour cela nous calculons les coordonnées physiques du point d'intégration ainsi que la normale en ce point. Puis à partir des coordonnées du point d'intégration sur la face, nous calculons dans la maille R les coordonnées de référence de ce point. Enfin par le même procédé que pour la maille L , nous calculons les champs en ce point. Toutefois, cette extrapolation requiert la lecture en mémoire globale des champs contenus dans chacun des éléments voisins de L . Ces lectures ne sont évidemment pas coalescentes ce qui rend cette étape relativement lente.

Pour terminer cette étape, chaque *work-item* calcule le flux numérique au point qui lui est associé puis écrit le résultat en mémoire locale dans le tableau qui, pour l'instant, contenait les champs de l'élément L . Ces écritures ne doivent avoir lieu qu'une fois ces champs inutilisés. Il faut donc demander une synchronisation des *work-items* appartenant à un même *work-group*. Une autre synchronisation est ajoutée après ces écritures, assurant la disponibilité des flux à tous les *work-items*.

Nous pouvons alors effectuer la seconde étape du calcul. Lors de cette étape, nous répartissons les *work-items* sur les points d'intégration volumiques, c'est-à-dire que le *work-item* d'indice j calculera l'intégrale surfacique pour la fonction de base d'indice $i = j$.

Pour cela, une boucle sur les points d'intégration des faces est effectuée et à chaque étape de cette boucle un terme de la somme (5.2 page précédente) est ajouté dans un tableau. Enfin, nous écrivons le résultat en mémoire globale dans le vecteur contenant les dérivés en temps de champs. La répartition des

work-items, nous assure la coalescence de ces accès mémoires.

Terme de masse

Dans ce *kernel*, nous calculons le terme de masse, puis nous divisons les champs par celui-ci. La matrice de masse étant diagonale la parallélisation se fait naturellement en associant à chaque *work-items* une fonction de base. Dans un premier temps, nous calculons le terme de masse associé à cette fonction de base (4.31 page 73). Puis nous divisons la composante correspondante du champ par cette valeur.

À nouveau, ces calculs font intervenir des fonctions qui dépendent de la géométrie et de l'interpolation.

5.1.4 Optimisation sur maillage hexaédrique

Dans cette section, nous supposons que les mailles présentes sont toutes des hexaèdres. Cette supposition nous permet de simplifier les formules donnant les intégrales surfaciques et volumiques ainsi que le calcul des champs sur les faces d'une maille. Le degré d'approximation est noté d . Cet entier est constant dans chaque zone, mais peut varier d'une zone à l'autre.

En pratique, nous n'avons programmé que la résolution sur maillage hexaédrique. La méthode générique, présentée dans la section précédente, a servi à valider les résultats et à quantifier les performances de la méthode optimisée. Celle-ci nous laisse également la possibilité d'ajouter la prise en charge d'autres types de maille.

Calcul des termes volumiques

Nous calculons dans ce *kernel*, l'intégrale (4.35 page 74) en tirant parti de la propriété vérifiée par le gradient des fonctions de base exposée dans la section 4.2.2 et tirée de [16]. Pour un indice i fixé, nous évaluons cette intégrale en ne considérant que les points alignés avec le point i . Observons alors les termes de cette somme

$$\sum_j \omega_j F(W_j, W_j, \text{com}(\tau'_L)(\hat{x}_j) \hat{\nabla} \hat{\psi}_i(\hat{x}_j)). \quad (5.4)$$

Les termes demandant le plus de calcul sont le flux F et la comatrice $\text{com}(\tau'_L)(\hat{x}_j)$. Le flux dépend des deux indices i et j , alors que la comatrice ne dépend que de l'indice j . Nous parallélisons donc ce calcul en attribuant à un *work-item* le calcul des termes pour un indice j . Ainsi, nous n'avons à calculer qu'une seule fois la comatrice $\text{com}(\tau'_L)(\hat{x}_j)$.

Un *work-item* doit alors ajouter les contributions qu'il calcule au tableau contenant la dérivée en temps pour différentes valeurs de l'indice i . Nous devons alors prendre garde à ne pas provoquer de conflit d'écriture, c'est-à-dire assurer que deux *work-items* n'écrivent pas au même endroit au même moment. Pour cela, nous allons ajouter les contributions sur des indices i différents pour chaque *work-item*.

Soit p, q et r trois entiers compris entre 0 et d tels que $j = p + (d+1)(q + (d+1)r)$. Soit j' un second indice correspondant à un autre *work-item* s'écrivant $j' = p' + (d+1)(q' + (d+1)r')$ avec p', q' et r' compris entre 0 et d . Notons R_d l'application qui à un entier associe le reste de la division euclidienne de cet entier par $d+1$. Définissons alors six indices

$$\begin{aligned}
i_0(\alpha) &= R_d(p + \alpha) + (d+1)(q + (d+1)r), \\
i_1(\alpha) &= p + (d+1)(R_d(q + \alpha) + (d+1)r), \\
i_2(\alpha) &= p + (d+1)(q + (d+1)R_d(r + \alpha)), \\
i'_0(\alpha) &= R_d(p' + \alpha) + (d+1)(q' + (d+1)r'), \\
i'_1(\alpha) &= p' + (d+1)(R_d(q' + \alpha) + (d+1)r'), \\
i'_2(\alpha) &= p' + (d+1)(q' + (d+1)R_d(r' + \alpha)),
\end{aligned} \tag{5.5}$$

pour α compris entre 0 et d . Remarquons tout d'abord que les indices i_0, i_1 et i_2 parcourent tous les indices i alignés avec j . De plus, pour α fixé et tout $n = 0, 1, 2$, les indice $i_n(\alpha)$ et $i'_n(\alpha)$ ne sont pas égaux. Donc, les indices i_0, i_1 et i_2 définissent une séquence pour laquelle aucun conflit d'écriture n'apparaît et qui permet de parcourir tous les points i nécessaires au calcul de la somme (4.35 page 74).

Toutefois, le parcours des indices i ainsi défini passe trois fois par l'indice central j , en effet $i_0(0) = i_1(0) = i_2(0) = j$. Nous ne commençons donc ces séquences qu'en $\alpha = 1$ et ajoutons le terme correspondant à j une seule fois.

L'algorithme obtenu en appliquant la procédure décrite précédemment est donné en pseudo-code en 5.2. Celui-ci est parallélisé en associant un *work-group* à un élément. Chaque *work-group* est composé de $(d+1)^3$ *work-items* étant associé chacun à un indice j .

```

L = indice du work-group
j = indice du work-item

// Termes constants
wj = poids j
codtau = comatrice au point j

// Lecture des champs en memoire globale
WL = champs au point j de L

gradpsi = gradient j au point j

// Ajout des contributions en memoire locale
local_dtw[j] = wj * F(WL, WL, codtau * gradpsi)

// Point de rencontre des work-items
Synchronisation locale

for (a=1;a<d+1;a++){
  for (n=0;n<3;n++){
    i = i_n(a);
    gradpsi = gradient i au point j

    // Ajout des contributions en memoire locale
    local_dtw[i] += wj * F(WL, WL, codtau * gradpsi)

    // Point de rencontre des work-items
    Synchronisation locale
  }
}

dtw[j] = local_dtw[j];

```

Algorithme 5.2 – Algorithme parallèle de calcul des intégrales volumiques.

Plusieurs remarques peuvent être effectuées sur cet algorithme :

Remarque 13. *Au début de la procédure, nous lisons les champs associés au point j en mémoire globale. L'efficacité de cette lecture est assurée par l'ordre de rangement des données des champs en mémoire. De la même manière, le résultat est écrit en mémoire globale en fin de calcul. À nouveau, cette écriture est effectuée de manière coalescente.*

Remarque 14. *Les contributions sont additionnées en mémoire locale, cette*

mémoire est d'accès rapide, mais ne perdure pas après la fin de l'exécution du kernel. Ainsi, l'association d'un work-group à un élément est essentielle.

Remarque 15. *L'utilisation de la mémoire locale est impérative. Dans le cas contraire, un grand nombre d'accès non coalescents à la mémoire globale serait effectué, dégradant fortement les performances. Par exemple, au degré 2 avec lorsque $\alpha = 1$, les work-items associés aux indices 0, 1 et 2 accèdent aux valeurs d'indices 1, 2 et 0.*

Remarque 16. *Après chaque écriture en mémoire locale les work-items sont synchronisés. Cette synchronisation assure que tous les calculs utiles pour la suite sont terminés.*

Remarque 17. *Au cours des itérations sur les points i , nous ne calculons pas toutes les composantes du gradient de la fonction de base. En effet, puisque i est différent de j , seule la composante correspondant à la direction d'alignement de i et j , c'est-à-dire n , est non nulle.*

Calcul des termes surfaciques

En supposant que les mailles sont hexaédriques, plusieurs simplifications apparaissent dans la formulation des intégrales surfaciques. Nous utiliserons la même stratégie de parallélisation que dans le cas générique, *i.e.* la parallélisation est effectuée sur les éléments et les intégrales surfaciques sont calculées deux fois. De même, le calcul est séparé en deux étapes similaires au cas générique.

Les premières simplifications apparaissent dans le calcul des champs surfaciques W_L provenant de l'élément courant. Ces champs sont extrapolés à partir des champs connus dans le volume sur les points d'intégration des faces. Afin d'obtenir le champ W_L au point d'intégration $x_{f,j'}$ de la face f , nous n'utiliserons que les contributions associées aux points d'interpolation alignées avec le point $x_{f,j'}$. Les indices de ces points sont les antécédents de j' par l'application $j \mapsto j' = \pi(f, j)$ définie en (4.13 page 66) et peuvent être déterminé simplement par des divisions euclidiennes.

Le calcul des champs W_R provenant des éléments voisins est effectué suivant le contexte. Dans le cas où l'ordre d'approximation est constant dans tout le maillage, nous pouvons appliquer la même procédure de calcul que pour les champs W_L puisque le point d'intégration $x_{f,j'}$ correspond à un point d'intégration surfacique de l'élément R . Pour cela, nous commençons par déterminer les indices de la face et du point d'intégration dans R . Ces indices sont déterminés en utilisant les données de connectivité des éléments et

en tenant compte des orientations des faces. Nous appliquons alors la procédure permettant d'extrapoler le champ W_R en utilisant les points volumiques alignés avec le point surfacique. Cette optimisation est très intéressante puisqu'elle nécessite la lecture de $d + 1$ champs en mémoire globale au lieu de $(d + 1)^3$. Comme ces lectures ne sont pas coalescentes, limiter leur nombre permet un gain de temps significatif.

Toutefois, nous ne pouvons appliquer cette astuce qu'en cas d'une approximation d'ordre constant dans tout le maillage. En effet, les calculs sont effectués au moyen de *kernels* communs à toutes les zones et doivent donc pouvoir prendre en compte tous les cas de figure rencontrés. Ainsi, si l'ordre d'approximation varie, nous calculons les champs W_R en utilisant la procédure décrite en 5.1.3.

Comme précédemment, une fois les champs W_L et W_R calculés, chaque *work-item* calcule le flux numérique au point qui lui est associé et l'écrit en mémoire locale.

Les *work-items* sont alors répartis sur les points d'interpolation du volume comme dans le cas générique. Le *work-item* d'indice i calcule l'intégrale (4.44 page 77) pour ce même indice i . Comme précédemment, les *work-items* d'indice $i \geq (d + 1)^3$ sont inactifs. En appliquant la propriété 11 page 68, l'estimation de l'intégrale (4.44 page 77) ne nécessite la contribution que d'un flux par face. Pour une face f donnée, ce flux est associé au point d'intégration surfacique sur lequel le point volumique i est projeté. L'indice i' de ce point est donné par l'application π définie en (4.13 page 66), soit $i' = \pi(f, i)$.

Ceci conduit à l'algorithme suivant 5.3.

```

L = indice du work-group
i = indice du work-item

// Seuls les work-items d'indice inferieur au nombre
// de points d'integration surfaciques sont actifs.
if(i < 6*(d+1)^2){
    // Calcul des indices j' et f en fonction de i
    jprime, f := (i = jprime + f * (d+1)^2)

    // Chargement en memoire locale des champs de L
    WL = Champs en jprime de L

    // Lecture de la connectivite

```



```

R = voisin de L par la face f

// Calcul en memoire globale de wr
WR = champs en jprime de R

// calcul de la normale et du poids
wj = poids surfacique jprime
Vds = normale a la face f en jprime

local_flux[f, jprime] = F(Wl, WR, Vds)
}

// Synchronisation des work-items
Synchronisation locale

// Seuls les work-items d'indice inferieur au nombre
// de points d'integration volumiques sont actifs.
if(i < (d+1)^3){
// Repartition des work-items sur les
// points volumiques
dtw = 0
for(f=0;f<6;f++){
dtw += local_flux[f, pi(f, i)]
}

// Ecriture du resultat en memoire globale
global_dtw[i] += dtw
}

```

Algorithme 5.3 – Algorithme parallèle de calcul des intégrales surfaciques.

Dans ce qui précède, nous avons présenté les algorithmes parallèles OpenCL. Deux versions ont été programmées : une version générique et une version optimisée pour un maillage hexaédrique présentant une interpolation constante. Les *kernels* ainsi programmés sont adaptés aux périphériques de calcul de type GPU. En particulier, nous avons cherché à économiser la mémoire, préférant recalculer certaines valeurs constantes plutôt que de les stocker. Toutefois, la taille de la scène pouvant être simulée reste limitée. Afin de pouvoir simuler des scènes de grande taille, nous allons paralléliser les calculs sur plusieurs périphériques OpenCL en utilisant la bibliothèque MPI.

5.2 Parallélisation MPI

Dans cette section, nous présentons la stratégie mise en oeuvre pour paralléliser les calculs sur plusieurs périphériques OpenCL en utilisant la bibliothèque MPI. Cette bibliothèque permet de réaliser des calculs sur des machines à mémoire distribuée. Plusieurs processus sont lancés en parallèle. Chacun possède son propre espace mémoire. Les différents processus communiquent alors entre eux par un système de message.

L'approche décrite ici est basée sur une technique de sous-domaine. Le maillage global est réparti entre les différents processus en prenant garde à équilibrer le volume de calcul entre les différents processus MPI. Chaque processus va alors utiliser un périphérique OpenCL différent. Les mises à jour entre processus sont nécessaires lors du calcul des termes surfaciques. Ceux-ci nécessitent la connaissance des champs provenant des mailles voisines. Si la maille voisine est traitée par un processus différent, nous devons récupérer cette information.

Il faut donc transférer entre processus des données volumiques. Pour cela, une couche de mailles provenant du maillage voisin est partagée entre deux sous-domaines voisins. La figure [FIGURE 5.2](#) page suivante représente le recouvrement obtenu dans le cas de deux processus. Les maillages traités par les deux processus présentent alors un recouvrement de deux couches de mailles. Les champs électromagnétiques des couches communes sont échangés deux fois par pas de temps puisque nous effectuons une intégration en temps de Runge-Kutta d'ordre 2.

La répartition des mailles en zones tient alors compte du « statut MPI » des mailles (interne, à envoyer ou à recevoir). Une zone ne devra alors contenir que des mailles ayant le même statut MPI. Cette répartition permet le transfert entre le périphérique de calcul et le périphérique hôte d'un bloc de données plutôt que de données éparses.

Avant le calcul des termes surfaciques, deux processus traitant des domaines voisins vont alors s'envoyer les données portées par la couche de mailles qu'ils ont calculées et recevoir celles de la couche traitée par le processus voisin. Cet échange de données est effectué de manière synchrone, c'est-à-dire que durant l'échange de données aucune autre tâche n'est effectuée. Afin de réaliser ce transfert, les données des zones à envoyer sont transférées du périphérique de calcul à l'hôte. Ensuite, ces données sont organisées de manière à être compréhensibles par les deux processus. Pour cela, nous conservons l'indice des mailles dans le maillage initial et ordonnons les données associées à chaque maille suivant cet indice. Nous transférons alors les données entre

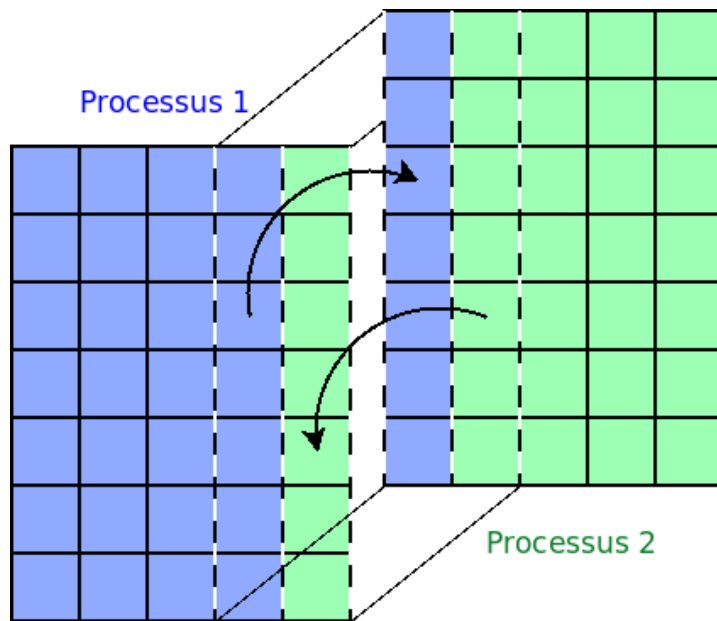


FIGURE 5.2 – Recouvrement de maillage de deux processus MPI. Les flèches symbolisent les envois de données.

les processus. Les données reçues sont alors ordonnées suivant les indices des éléments dans la partie du maillage courant.

Les calculs surfaciques sont alors effectués de la même manière que précédemment excepté dans la couche supplémentaire, reçue du voisin, où aucun calcul n'est effectué.

Cette approche requiert le stockage de données volumiques supplémentaires ainsi que leur transfert. De plus, les données nécessitent d'être réorganisées avant le transfert. Nous discuterons l'efficacité de cette approche dans le test décrit en 6.5. Nous proposerons également une alternative permettant de réduire le coût de transfert dans le chapitre 7.

Conclusion

En conclusion de ce chapitre, nous avons donc programmé la méthode GD sur des hexaèdres. La méthode est parallélisée en utilisant l'environnement OpenCL pour piloter les calculs qui se déroulent sur un GPU. Nous utilisons la librairie MPI pour distribuer les calculs sur plusieurs GPU. Pour l'instant, les échanges entre les GPU sont synchrones. Nous avons programmé plusieurs

versions de chaque *kernel* : une version générique, qui pourrait fonctionner pour tout type d'élément de référence, comme des tétraèdres ou des prismes ; une version optimisée pour les hexaèdres.

Dans le chapitre suivant, nous allons valider cette programmation sur différents cas-tests académiques ou plus physiques. Nous allons aussi estimer les performances de notre approche.

Chapitre 6

Validations

Dans ce chapitre, nous présenterons les cas tests mis en place afin de valider la méthode et les différents modèles physiques présentés dans les chapitres précédents. Les cas tests sont construits afin de valider un ensemble réduit, mais significatif, de modèles. Nous prendrons comme référence une solution analytique lorsque nous en disposons ou les résultats obtenus à l'aide d'un logiciel de référence. Les différents calculs sont effectués en simple précision.

Le premier cas test a pour but de valider le coeur de calcul GD en comparant l'ordre de convergence de la méthode obtenu numériquement avec l'ordre de convergence théorique. Le second cas test permet de mettre en évidence la validité fréquentielle de la méthode, c'est-à-dire d'estimer en fonction du pas de discrétisation spatiale une fréquence maximale pouvant être correctement représentée.

Les cas tests suivants font partie des tests de validation du projet GREAT (Galerkin Resolution for Electromagnetic Applications in Time domain, financé par DGA/DS/MRIS dans le cadre d'un projet d'innovation duale RAPID). Ces tests visent à valider la méthode sur des cas d'application et à estimer l'efficacité du code. Nous considérons pour cela différentes configurations représentatives de simulation que nous pourrions être amenés à effectuer avec la méthode GD. Nous présenterons ici trois de ces cas tests.

6.1 Ordre de convergence de la méthode

Dans cette section, nous allons vérifier que nous obtenons numériquement un ordre de convergence de cette méthode pour des degrés d'approximation

$d = 0, \dots, 4$ qui correspondent à l'ordre de convergence théorique de la méthode. Nous nous attendons à une erreur d'ordre $d + 1$ en norme L^2 . La démonstration de convergence présentée dans la section 2.6 page 41 donne un ordre $d + 1/2$, mais pour une norme plus fine contenant un terme de saut sur les discontinuités.

Pour cela, considérons un hexaèdre déformé dont les nœuds sont reportés en (6.1) et représenté sur la figure FIGURE 6.1 page suivante. Nous injectons alors une solution exacte des équations de Maxwell dans ce domaine de calcul au moyen d'une condition aux limites de type Silver-Müller inhomogène. Nous choisissons ici une onde plane monochromatique se propageant dans la direction $V = (-0.0361, 0.170, 0.985)^T$ et de fréquence $f = 600MHz$.

$$\begin{aligned}
S_0 &= (0., 0., 0.) \\
S_1 &= (1., 0., 0.) \\
S_2 &= (0.866, 1., 0.5) \\
S_3 &= (0., 1., 0.) \\
S_4 &= (0., 0., 1.) \\
S_5 &= (0.866, 0.5, 1.) \\
S_6 &= (-0.183, 1.37, 1.05) \\
S_7 &= (-0.933, 0.866, 0.616)
\end{aligned} \tag{6.1}$$

Les champs électromagnétiques sont alors donnés par

$$\mathcal{E}(X, t) = \sin(2\pi f(tc - V \cdot X)) \begin{pmatrix} 0.752 \\ -0.644 \\ -0.139 \end{pmatrix} \tag{6.2}$$

et

$$\mathcal{H}(X, t) = \sin(2\pi f(tc - V \cdot X)) \begin{pmatrix} 0.658 \\ 0.746 \\ 0.105 \end{pmatrix} \tag{6.3}$$

Nous effectuons le calcul sur une durée de $2ns$ avec un pas de temps fixé assurant la stabilité du schéma à chaque degré d'approximation pour tous les raffinements que nous considérerons. Ce pas de temps satisfait une CFL très faible afin de supprimer l'erreur due à l'intégration en temps, qui n'est que d'ordre 2.

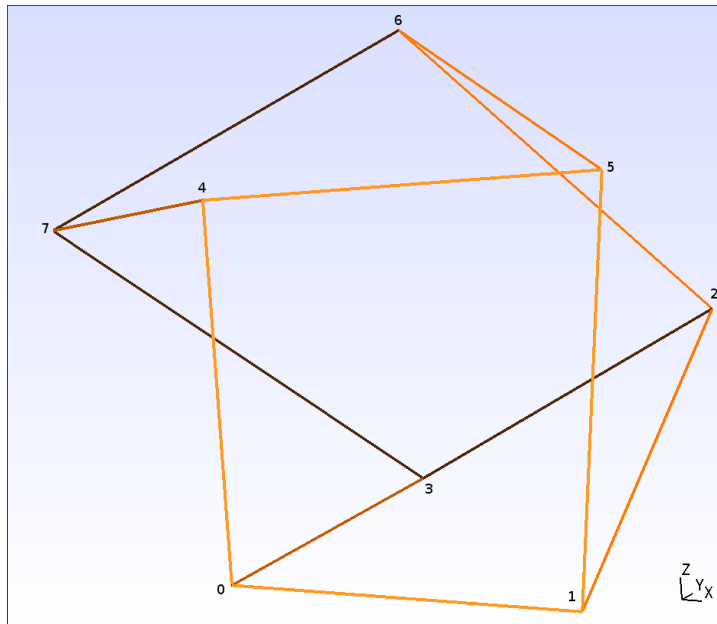


FIGURE 6.1 – Géométrie du domaine de calcul hexaédrique dont les coordonnées des nœuds sont données en (6.1 page précédente) et utilisé pour l'étude de convergence.

La figure FIGURE 6.2 page suivante représente le champ électrique sur une face du cube unité à l'instant final. Nous pouvons observer que dans cette configuration, les champs se propagent de différentes manières dans les éléments : suivant une direction cardinale ou suivant un axe oblique.

Nous effectuons le calcul pour les différents degrés d'approximation et calculons la norme L^2 de la différence entre la solution numérique et la solution analytique. La figure FIGURE 6.3 page suivante représente en échelle logarithmique l'évolution de l'erreur en fonction du pas de discrétisation ainsi que les courbes étalons de pente théorique.

Nous obtenons bien des ordres de convergence compatibles avec les ordres théoriques. Le tableau 6.1 contient les ordres numériques et théoriques.

Comparons de plus les temps de calcul obtenus avec les implémentations générique et optimisée pour un maillage hexaédrique des *kernels*. Pour cela, nous effectuons les simulations précédentes avec les deux implémentations. Nous construisons un maillage constitué de $30^3 = 27000$ mailles afin d'avoir une charge correcte du GPU. Les temps de calcul ainsi obtenus sont reportés dans le tableau 6.2. Nous obtenons alors une accélération d'un facteur 3 entre l'implémentation générique et l'implémentation optimisée.

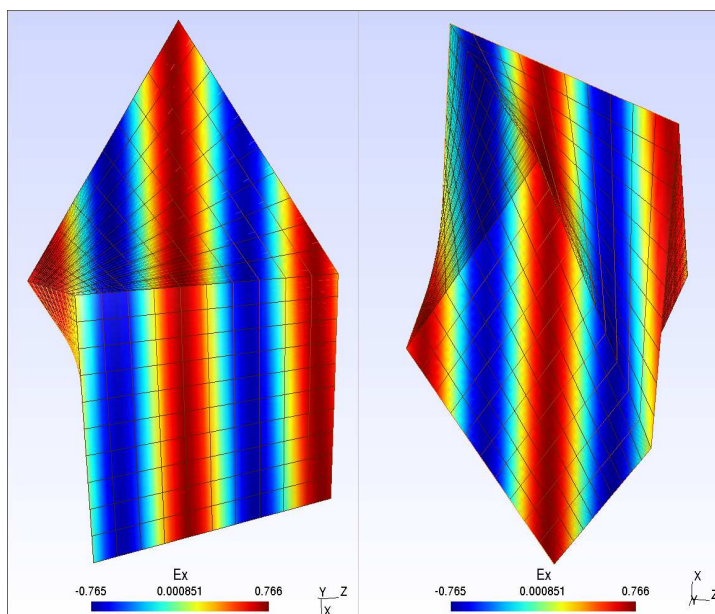


FIGURE 6.2 – Composante E_x du champ électrique à $t = 2$ ns vue sous deux angles différents.

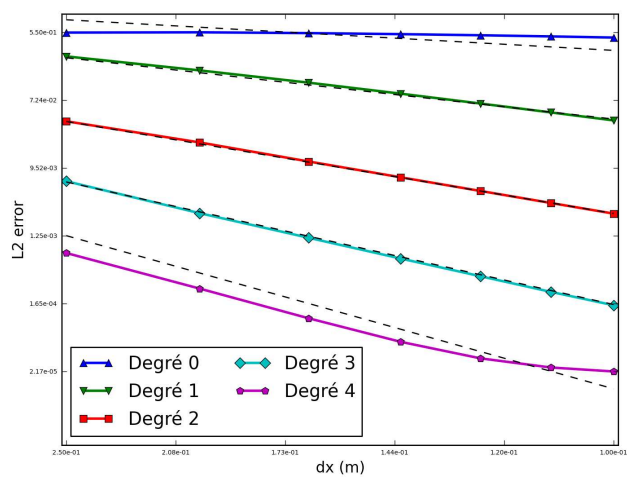


FIGURE 6.3 – Évolution de l'erreur en fonction du pas de discrétisation (échelle logarithmique). Degré 0 : triangles bleus. Degré 1 : triangles inversés verts. Degré 2 : carrés rouges. Degré 3 : losanges cyan. Degré 4 : pentagones violets.

Degré	Ordre théorique	Ordre numérique
0	1	0.3
1	2	2.12
2	3	3.01
3	4	3.97
4	5	4.48

TABLE 6.1 – Ordres de convergence théoriques et numériques pour différents ordres d’approximation.

Degré	0	1	2	3	4
<i>kernels</i> génériques (ms)	33.9	42.8	58.6	92.2	227
<i>kernels</i> optimisés (ms)	11.3	15.0	18.5	30.0	70.8
Accélération	3.0	2.9	3.2	3.1	3.2

TABLE 6.2 – Temps de calcul par itération obtenus avec les *kernels* génériques et optimisés sur un maillage constitué de 27000 éléments.

6.2 Domaine de validité fréquentielle

Dans cette section, nous étudions le spectre fréquentiel pouvant être représenté pour un pas de maillage donné. En effet, le théorème de Shannon stipule que la représentation discrète d’un signal nécessite un échantillonnage de fréquence deux fois supérieure à la fréquence maximale du signal. En pratique, il est nécessaire de choisir un échantillonnage plus fin afin de représenter correctement un tel signal. Dans l’utilisation de méthode type différences finies en domaine temporel (FDTD) appliqué à l’électromagnétisme, il est courant d’appliquer la règle dite « λ sur 10 ». Celle-ci permet de déterminer un pas de maillage (*i.e.* une fréquence d’échantillonnage spatiale) en fonction de la longueur d’onde minimale observable.

Par analogie, nous allons déterminer pour différents degrés d’approximation une règle similaire applicable à une simulation GD. Nous appellerons ici pas de maillage et noterons h la distance minimale séparant deux nœuds du maillage. Ce pas correspond au pas FDTD dans le cas d’un maillage composé de cubes.

Considérons un maillage composé de cubes dont les arêtes mesurent 0.1m. Le pas de maillage vaut alors $h = 0.1$ m. Nous injectons dans le domaine de calcul une onde plane via une surfacique de Huygens. Cette condition surfacique est décrite dans la section 3.6 page 57. Nous choisissons une forme d’onde de

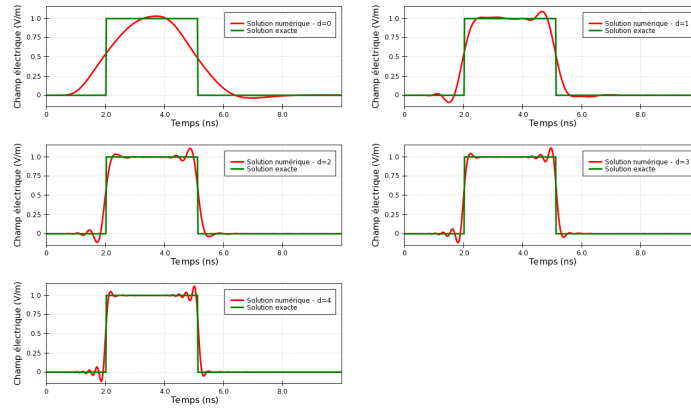


FIGURE 6.4 – Composante majoritaire du champ électrique au centre du domaine aux différents degrés d’approximation. Des oscillations de Gibbs sont visibles au niveau des discontinuités de la solution. Avec la montée en ordre, les oscillations ont tendance à se resserrer.

type échelon définie par

$$g(t) = \begin{cases} 1 & , \text{ si } t \in [3.5 \times 10^{-10}, 3.5 \times 10^{-9}] \\ 0 & , \text{ sinon.} \end{cases} \quad (6.4)$$

Nous choisissons cette forme d’onde pour deux raisons. Son support est fini, ainsi nous pouvons calculer des transformées de Fourier du signal calculé. Son spectre est infini et son amplitude décroît lentement (en $1/f$) contrairement à une forme d’onde gaussienne dont la décroissance en fréquence est exponentielle. Pour cette forme d’onde, la solution d’onde plane présente deux discontinuités qui se propagent dans le domaine. Ce cas de figure n’est pas physique et ne sera donc pas rencontré en pratique, mais ici nous souhaitons simplement déterminer jusqu’à quelle fréquence l’approximation reste correcte.

Nous effectuons cette simulation avec des degrés d’approximation allant de 0 à 4. Nous relevons le champ électrique au centre du cube. La présence de discontinuités dans la solution exacte entraîne l’apparition d’oscillations dans la solution numérique. Ces oscillations sont assimilables à un phénomène de Gibbs. La figure FIGURE 6.4 représente les champs obtenus ainsi que la solution exacte.

Comparons les contenus fréquentiels de ces signaux. Les figures FIGURE 6.5a page 116 et FIGURE 6.5b page 116 représentent les modules des transformés de Fourier

des signaux au centre du cube. Ne considérons alors que les valeurs situées aux maxima locaux et normalisons par rapport au signal envoyé. Le rapport obtenu donne la fraction de champs émis à une fréquence qui s'est effectivement propagée. La figure FIGURE 6.6 page 117 représente ces rapports en fonction de la fréquence.

En fixant un seuil (choisi arbitrairement à 95% sur la figure FIGURE 6.6 page 117), nous pouvons déterminer quelle fréquence et donc quelle longueur d'onde a été simulée de façon acceptable et calculer le rapport entre h et cette longueur d'onde.

Pour des seuils de 95%, 98% et 99%, nous obtenons les rapports longueur d'onde sur pas d'espace donné dans le tableau 6.3.

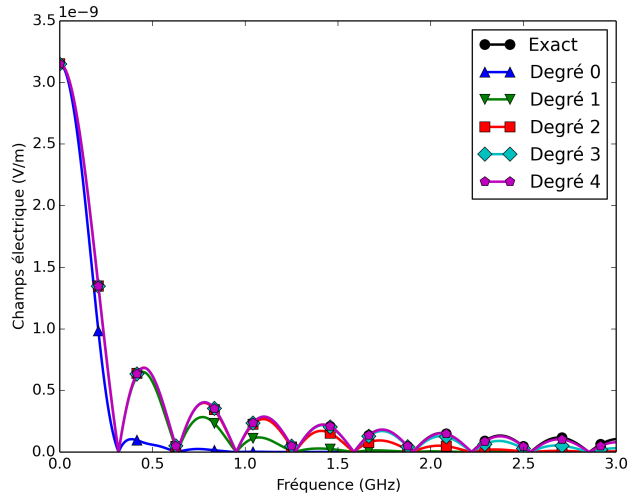
Seuil	Degré 0	Degré 1	Degré 2	Degré 3	Degré 4
95%	90	6	3	2	1.3
98%	200	17	3.6	2.2	1.4
99%	300	30	4.3	2.3	1.5

TABLE 6.3 – Rapport longueur d'onde observable sur pas d'espace pour différents degrés d'approximation et différents seuils de tolérance.

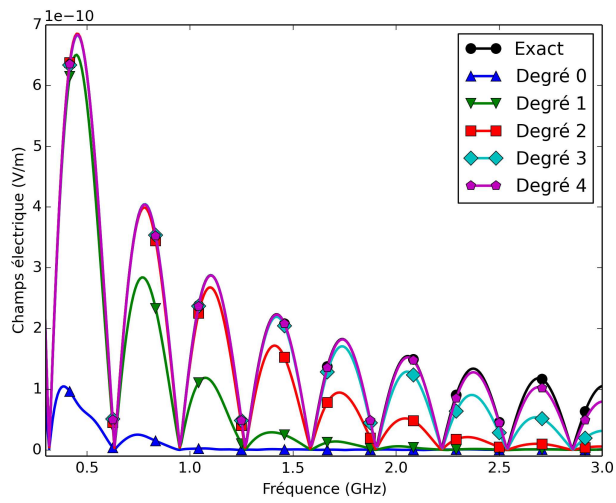
Il apparait que l'utilisation d'une approximation faite de fonction constante par morceaux (c'est-à-dire un ordre d'approximation $d = 0$) requiert un maillage très fin. Dans ce cas, la méthode GD devient une méthode volumes finis. La dispersion et la dissipation dues à ce type de méthode conduisent à des résultats de mauvaise qualité.

Pour des approximations de degré au moins deux des champs dans les mailles, nous obtenons une règle comparable à celle connue dans la simulation FDTD. Pour une approximation de degré d , une maille contient $d+1$ degrés de liberté dans chaque direction. En remplaçant le pas d'espace h par un pas tenant compte du degré d'approximation $h/(d + 1)$, nous obtenons des rapports allant de 7.5 à 12.9.

Il est à noter que la simulation d'une structure complexe requiert souvent d'appliquer une règle plus forte que celle obtenue ici. C'est également le cas lors de l'utilisation d'une méthode FDTD. Ceci est dû à la nécessité d'une prise en compte correcte des forts gradients de champ induits par la présence d'arêtes saillantes dans les structures et conditionnant fortement leur réponse. Certains travaux visent à une prise en meilleure prise en compte de ces phénomènes dans le cadre de la méthode FDTD, nous pouvons par exemple citer les travaux [59, 58] où une modification des paramètres physiques des matériaux permet cette prise en compte. Nous pouvons donc en-



(a) Contenus fréquentiels des signaux.



(b) Zoom sur la zone de décrochage.

FIGURE 6.5 – Contenu fréquentiel des signaux au centre du cube pour les différents degrés d'approximation la solution exacte.

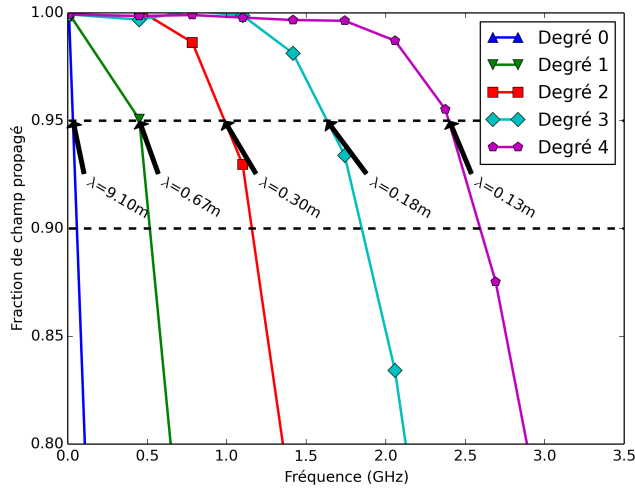


FIGURE 6.6 – Fraction de champs émis effectivement propagée en fonction de la fréquence et à différents degrés d’approximation.

visager de transcrire ce type d’approche en GD si cela s’avère nécessaire.

6.3 Ligne microruban

Ce cas test consiste en la simulation d’une ligne micro-ruban. Nous souhaitons calculer l’impédance de la ligne que nous comparerons au résultat d’un code FDTD ainsi qu’à un résultat théorique. Ce test est tiré de la thèse de Bouquet [6].

La géométrie du cas est représentée sur la figure FIGURE 6.7 page suivante et une vue du maillage GD est représentée sur la figure FIGURE 6.8 page suivante. La ligne micro-ruban est de largeur $w = 2.46\text{mm}$ et sa hauteur par rapport au plan de masse est $h = 0.795\text{mm}$. Notons z_0 et z_1 les coordonnées suivant l’axe z du plan de masse et le la ligne micro-ruban. Notons également x_0 la coordonnée suivant la direction x du milieu de la ligne micro-ruban. On injecte un champ incident sous la ligne micro-ruban via une surface de Huygens (voir la section 3.6 page 57). Le champ électrique est orienté suivant la direction z . Sa forme d’onde est une gaussienne modulée, donnée par la

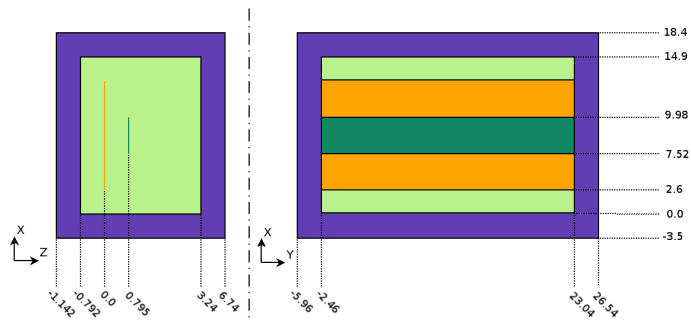


FIGURE 6.7 – Schématisation de la géométrie du cas test micro-ruban (unité : mm). Les PML sont représentées en violet, l'espace libre en vert clair, le plan de masse en orange et la ligne micro-ruban en vert foncé.

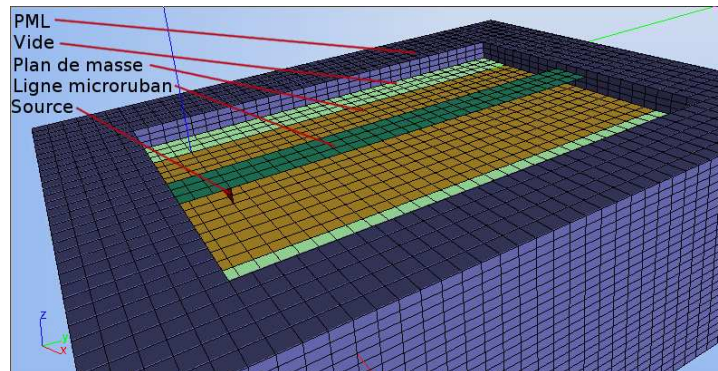


FIGURE 6.8 – Vue 3D du maillage de la ligne micro-ruban. Les PML sont représentées en violet, l'espace libre en vert clair, le plan de masse en orange et la ligne micro-ruban en vert foncé. Les PML et l'espace libre sont tronqués afin de voir le plan de masse et la ligne micro-ruban.

formule

$$g(t) = \sin(2\pi f_0(t - t_0)) \exp\left(-\left(\frac{t - t_0}{\tau}\right)^2\right). \quad (6.5)$$

Nous prenons comme paramètres $t_0 = 47ps$, $f_0 = 22GHz$ et $\tau = 22ps$.

Définissons alors les tension et courant sur la ligne micro-ruban. La tension $V(t, y_0)$ sous la ligne micro-ruban en $y = y_0$ à un instant t est égale définie par l'intégrale du champ électrique tangent sur un segment reliant la ligne micro-ruban au plan de masse, *i.e.*

$$V(t, y) = \int_{z_0}^{z_1} \mathcal{E}_z(t, x_0, y_0, z) dz. \quad (6.6)$$

Le courant $I(t, y)$ est défini par l'intégrale de la partie tangente du champ magnétique sur un contour entourant la ligne micro-ruban, *i.e.*

$$I(t, y_0) = \int_{C(y_0)} \mathcal{H} \cdot v dx \quad (6.7)$$

où $C(y_0)$ est un contour fermé inclus dans le plan $y = y_0$ entourant la ligne micro-ruban et v est un vecteur unitaire tangent à $C(y_0)$.

Nous calculons les tensions V et courants I sur la ligne micro-ruban en différents plans orthogonaux à la ligne. Ces plans sont placés à des coordonnées y valant $2w = 4.92mm$, $5w = 12.3mm$, $6w = 14.76mm$ et $7w = 17.22mm$.

Le maillage est constitué de 56416 hexaèdres pour un pas d'espace de d'environ 0.397mm. Nous avons alors 2 mailles sous la ligne micro-ruban et 3 mailles dans sa largeur. Nous effectuons la simulation en prenant un degré d'approximation valant 2. Le domaine de calcul est entouré de couches absorbantes PML.

Sur la figure **FIGURE 6.9** page suivante, sont représentées les tensions en fonction du temps aux différentes abscisses d'observation. Nous remarquons que la tension décroît en s'éloignant de la source pour se stabiliser à partir d'une distance d'environ trois fois la largeur de la ligne. Ce phénomène est dû à la nature de l'excitation utilisée. En effet, nous injectons sous la ligne un champ incident grâce à une surface de Huygens. Ainsi ce champ à une

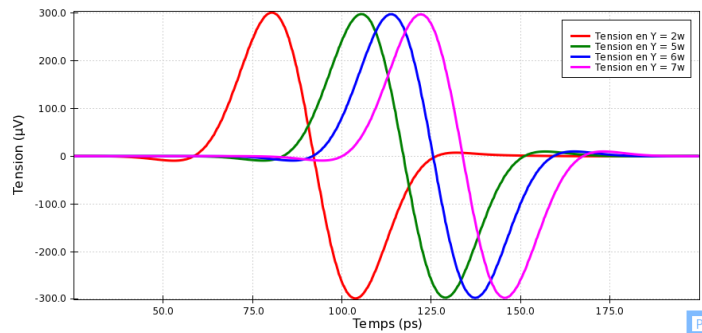


FIGURE 6.9 – Tensions sous la ligne micro-ruban à différentes distances de la source. On observe une légère décroissance puis une stabilisation de la tension due à l'établissement du champ propagatif.

forme d'onde plane et engendre donc sous la ligne des ondes évanescentes qui demandent une certaine distance pour être dissipées.

Nous calculons alors l'impédance de la ligne micro-ruban $Z = V/I$ à une abscisse où les tensions et courants sont stabilisés. L'impédance de la ligne vaut théoriquement 68.72Ω . Une formule permettant de calculer une valeur approchée de l'impédance de la ligne est donnée dans [17], section 9.3.1. Nous obtenons numériquement une impédance de 68.4Ω soit une erreur relative de l'ordre de 1%.

Afin d'avoir un point de comparaison, nous avons effectué la simulation de ce cas avec une méthode FDTD en choisissant une discrétisation relativement fine de l'espace formée de $253 \times 103 \times 57$. Dans cette configuration, l'espace sous la ligne micro-ruban comprend 36 mailles dans la largeur de la ligne et 12 mailles dans la hauteur. Les résultats de ce test sont récapitulés dans le tableau 6.4.

Méthode	FDTD	GD
Impédance (Ω)	66.7	68.4
Erreur relative (%)	2.9	0.5
Million d'inconnues	15.9	16.4
Temps de calcul (s)	339	194

TABLE 6.4 – Tableau récapitulatif des résultats de la ligne micro-ruban.

Le schéma GD permet une bonne prise en compte de l'impédance de la ligne micro-ruban. Comme cité précédemment (voir la section 6.2 page 113) cela peut provenir de la mauvaise prise en compte des gradients forts de champ

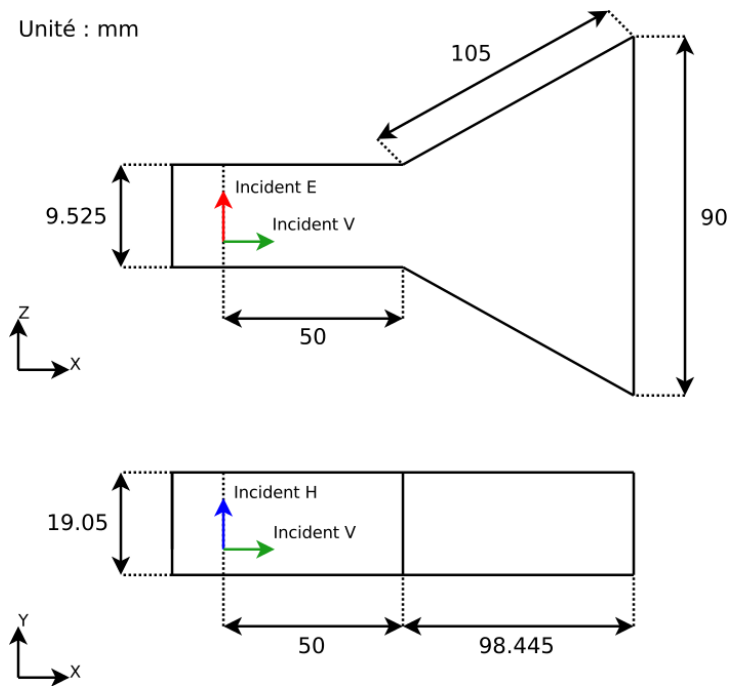


FIGURE 6.10 – Schéma de l'antenne cornet avec ses dimensions et l'orientation du champ incident.

engendrés par les arêtes de la structure. Dans le cas présent, la méthode FDTD utilisée n'utilise pas les modèles permettant la représentation fine des gradients de champs. Avec cette prise en compte, l'erreur serait diminuée.

Ce cas test a permis de valider l'interopération de différents modèles et de mettre en évidence l'apport de la méthode GD en terme de validité de solution due à une prise en compte fine des champs et structures, ainsi que son efficacité puisqu'elle conduit à des temps de calcul raisonnables pour une bonne précision.

6.4 Antenne cornet sectorale

Dans ce test, nous nous intéressons à la simulation d'une antenne rayonnante. L'antenne simulée est une antenne cornet sectorale dont les dimensions sont données sur la figure FIGURE 6.10. Nous calculons alors le ROS (Rapport d'Onde Stationnaire, nous donnerons plus loin sa définition) et le diagramme de rayonnement de l'antenne sur une bande fréquence allant de 7 à 22 GHz.

Afin de simuler un espace ouvert, le domaine de calcul est entouré de couches PML au nombre de 10. Le cornet est constitué de matériaux parfaitement conducteurs (voir 3.1). Dans ce cas de figure, nous avons constaté que la condition métal non dissipative ne garantit pas la stabilité du schéma. Cette instabilité semble due à la déformation des mailles sur les bords obliques du cornet. En appliquant une condition métallique dissipative, les instabilités disparaissent.

Le cornet est alimenté par un guide d'onde dont la section a pour dimension $19.05\text{mm} \times 9.525\text{mm}$. Ce guide est prolongé dans les PML afin de simuler une adaptation correcte du guide. Une surface rayonnante (voir 3.6) est introduite dans le guide d'onde. L'injection du champ incident via cette surface peut être effectuée en utilisant un profil compatible avec les dimensions du guide, par exemple en injectant un mode TE. Toutefois, cette approche présente plusieurs inconvénients. D'une part, ce type de source n'excite qu'une fréquence. D'autre part, le champ généré se propage dans les deux directions et son support temporel est infini. Ainsi, nous choisissons d'injecter le champ sous la forme d'une onde plane. Les champs électrique et magnétique incidents sont constants et tangents à la surface rayonnante ; le vecteur de propagation est orienté dans la direction du guide. La forme d'onde g est une gaussienne modulée par un sinus

$$g(t) = \sin(2\pi f_0(t - t_0)) \exp\left(-\left(\frac{t - t_0}{\tau}\right)^2\right) \quad (6.8)$$

avec $f_0 = 15\text{GHz}$, $t_0 = 0.205\text{ns}$ et $\tau = 60.4\text{ps}$. Ainsi, nous pouvons étudier le comportement fréquentiel du cornet sur la bande de fréquences souhaitée. Ce type de source introduit toutefois des ondes parasites dues à l'injection d'un champ inadapté à la structure du guide, néanmoins ces ondes sont évanescentes et le champ propagé adopte rapidement une structure TE.

Le maillage utilisé pour effectuer la simulation de ce cornet est constitué de 252 054 mailles munies d'une interpolation de degré 2, représenté sur la figure FIGURE 6.11 page suivante. Ce maillage est généré en construisant un maillage type FDTD de la structure puis en faisant subir une déformation aux éléments afin d'obtenir un maillage hexaédrique conforme à la structure. La simulation s'étend sur une durée de 3ns avec un pas de temps de 79.6fs soit environ 38 000 itérations.

Le ROS est calculé en relevant le champ électrique sur une ligne situé au milieu du guide alimentant l'antenne. Nous calculons alors le spectre fréquentiel de ce champ. À une certaine fréquence, Le ROS est donné par le rapport entre le minimum et le maximum de la norme du champ électrique

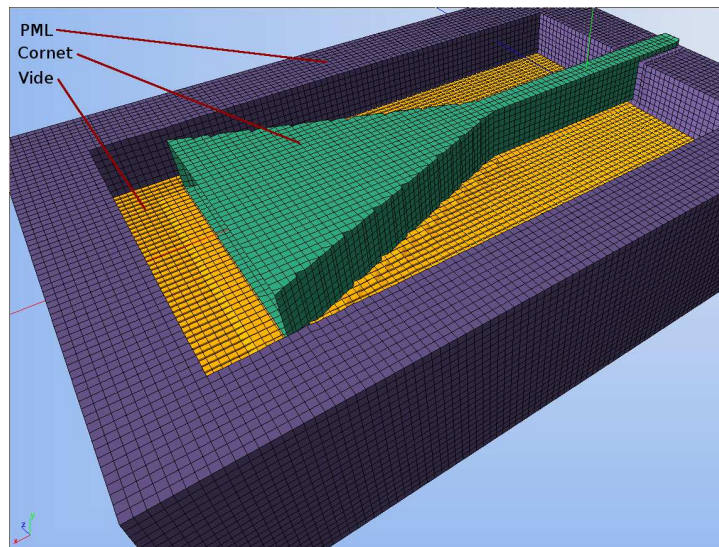


FIGURE 6.11 – Maillage de l’antenne cornet avec en violet les PML, en jaune l’espace libre et en vert le cornet. Des plans coupent les PML et l’espace libre afin de voir le cornet.

sur cette ligne. Le ROS traduit l’adaptation du guide d’alimentation à l’antenne (voir [17], chap. 3). Il est lié aux impédances de ces deux structures. Lorsqu’elles sont différentes, une partie de l’énergie injectée dans le système est réfléchi par le cornet. On peut également exprimer le ROS en fonction du taux de réflexion ρ entre le guide d’alimentation et l’antenne

$$\text{ROS} = \frac{1 + \rho}{1 - \rho}. \quad (6.9)$$

Anisi, plus le ROS est proche de 1 et plus l’adaptation est bonne.

Les résultats obtenus sont comparés avec ceux obtenus avec un outil FDTD développé par le CEA (Gorf3D). La figure FIGURE 6.12 page suivante représente les ROS obtenus avec la méthode GD et la méthode FDTD. Les niveaux sont corrects, mais on observe une disparité entre les deux courbes pouvant s’expliquer par le fait que la géométrie du cornet n’est pas exactement la même dans les deux cas (le maillage GD est conforme à la structure, mais pas le maillage FDTD). De plus, dans les hautes fréquences, les disparités peuvent être dues à des réflexions dans les PML que nous n’avons pas réussi à maîtriser.

La figure FIGURE 6.13 page 125 représente les diagrammes de rayonnement

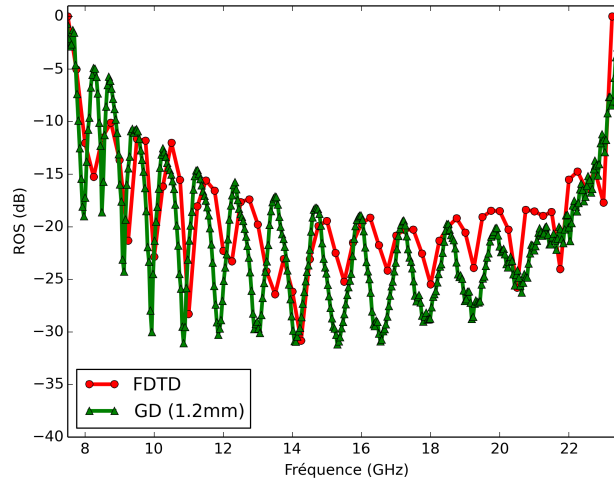


FIGURE 6.12 – ROS en fonction de la fréquence. En rouge des résultats obtenus avec les FDTD et en vert des résultats obtenus avec la méthode GD.

à un angle fixé. Les calculs ont été effectués avec la méthode FDTD et la méthode GD pour deux finesses de maillage correspondant au pas du maillage avant la déformation. Le résultat FDTD est représenté par les ronds rouges, le résultat GD 3mm par des carrés bleus et le résultat GD 1.2mm par des triangles verts. Les allures des diagrammes se correspondent bien et les niveaux des courbes semblent converger vers la courbe obtenue dans le cas le plus raffiné (GD 1.2mm). La figure FIGURE 6.14 page ci-contre représente le diagramme de rayonnement en 3D. On obtient une bonne concordance des résultats obtenus avec la méthode GD et la méthode FDTD.

6.5 Genec

Ce cas test consiste en la simulation d'un missile générique (GENEC). La géométrie de cette structure est représentée sur la figure FIGURE 6.15 page 127. Le cylindre constituant le GENEC a un rayon de 52.5mm et une longueur de 816mm.

D'une part, nous comparons les résultats calculés avec un code de référence et des mesures tirées de l'étude publiée en 2004 [67]. Pour cela, nous utilisons un maillage adapté à ce test. Le maillage est réalisé en tenant compte de la fréquence maximale observée (la discrétisation spatiale doit être assez fine

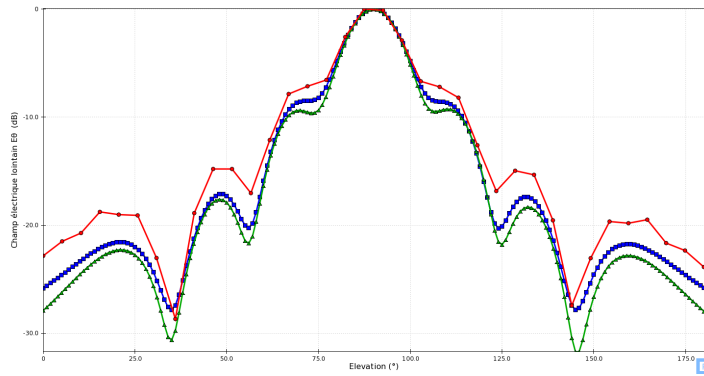


FIGURE 6.13 – Diagramme de rayonnement du cornet à 12GHz et $\theta = 0$. Les résultats proviennent d'un calcul FDTD (ronds rouges) et d'un calcul GD avec un pas de 3mm (carrés bleus) et 1.2mm (triangles verts).

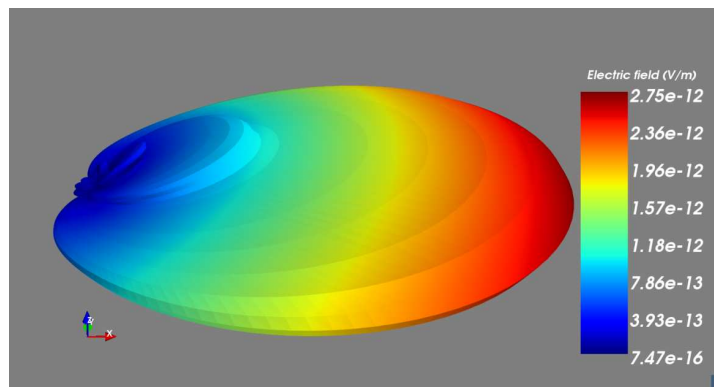


FIGURE 6.14 – Diagramme de rayonnement du cornet à 12GHz.

afin de pouvoir correctement représenter cette fréquence) et de la structure (les détails importants de la structure doivent être correctement représentés). Ce cas test a été réalisé dans le cadre du projet GREAT au CEA .

D'autre part, nous utilisons ce cas test afin d'estimer les performances de l'implémentation parallèle du schéma GD. Nous calculons pour cela la puissance de calcul de cette implémentation sur une, deux et quatre cartes graphiques. Dans chacun des cas, un maillage est réalisé afin de remplir la mémoire des périphériques de calcul. Pour cela, nous générons pour chacun des cas un maillage comprenant le nombre de mailles conduisant à la quantité de mémoire souhaitée. En supposant que le temps de calcul suit une loi linéaire, nous pouvons déduire des grandeurs calculées, l'accélération (« speedup ») obtenue lors de la parallélisation des calculs sur plusieurs cartes graphiques.

La structure est illuminée par une onde plane de forme d'onde gaussienne donnée par

$$g(t) = \exp\left(-\left(\frac{t-t_0}{\tau}\right)^2\right) \quad (6.10)$$

avec $t_0 = 2.9\text{ns}$ et $\tau = 59\text{ps}$ injectée dans le domaine de calcul grâce à une surface de Huygens (voir la section 3.6 page 57) entourant le GENECE. L'onde plane se propage dans la direction donnée par le vecteur $(-1, 0, 1)$ et le champ électrique est orienté suivant l'axe y .

6.5.1 Résultats

Afin de réaliser cette simulation, nous utilisons un maillage généré en réalisant un maillage tétraédrique dans la zone proche des structures, puis en découpant chaque tétraèdre en quatre hexaèdres. Par construction, les tétraèdres s'appuient sur un parallélépipède englobant la structure. Par extrusion des traces sur ce parallélépipède, un volume supplémentaire est ajouté au maillage, ce qui permet de raccorder le maillage aux couches PML. Ce maillage est représenté sur la figure FIGURE 6.15 page suivante. Sur cette figure, un plan de coupe permet de voir les points d'observations situés à l'intérieur du GENECE.

Nous calculons la fonction de transfert aux deux points de mesures de coordonnées $P_1 = (0.5755, 0, 0)$ et $P_2 = (0.4555, 0, 0)$ visibles sur la figure FIGURE 6.15 page ci-contre, et comparons les résultats obtenus avec les différentes méthodes et mesures [67]. La fonction de transfert $h(f)$ est définie

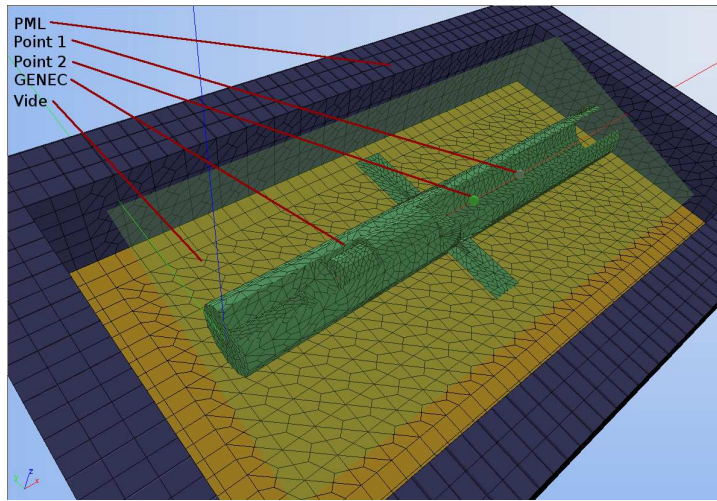


FIGURE 6.15 – Maillage du GENEC constitué de tétraèdres découpés en quatre hexaèdres et d’une extrusion. Les PML et l’espace libre sont tronqués afin de pouvoir observer la structure. Un plan de coupe permet de voir la position des points d’observation à l’intérieur du GENEC.

par le spectre temporel du champ $e(f)$ observé normalisé par le spectre de la source $s(f)$,

$$h(f) = \frac{e(f)}{s(f)}. \quad (6.11)$$

Les figures FIGURE 6.16 page suivante et FIGURE 6.17 page suivante représentent les différents résultats obtenus expérimentalement et par deux les méthodes numériques : FDTD et GD.

Nous observons une bonne concordance des résultats avec toutefois un léger décalage en fréquence. Les niveaux sont néanmoins respectés. Les fonctions de transfert obtenues avec l’algorithme GD parallèle concordent avec les résultats de référence.

6.5.2 Performances

Nous effectuons la simulation présentée précédemment en utilisant une, deux et quatre cartes graphiques NVIDIA GTX 780 Ti. Nous utilisons pour chacun des calculs un maillage différent. La taille des maillages est choisie afin de remplir la mémoire de chaque GPU. Nous ne considérerons pas ici l’interpo-

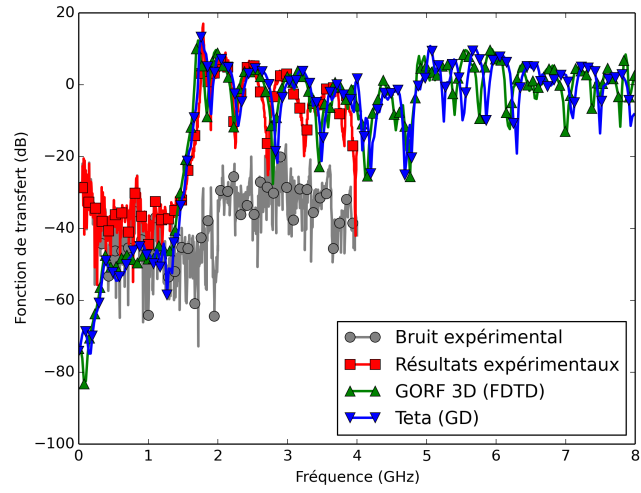


FIGURE 6.16 – Fonction de transfert au point 1 situé à l'intérieur du GENEK.

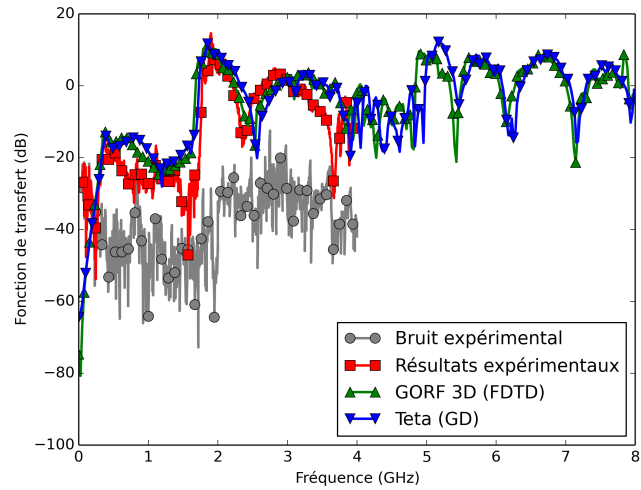


FIGURE 6.17 – Fonction de transfert au point 2 situé à l'intérieur du GENEK.

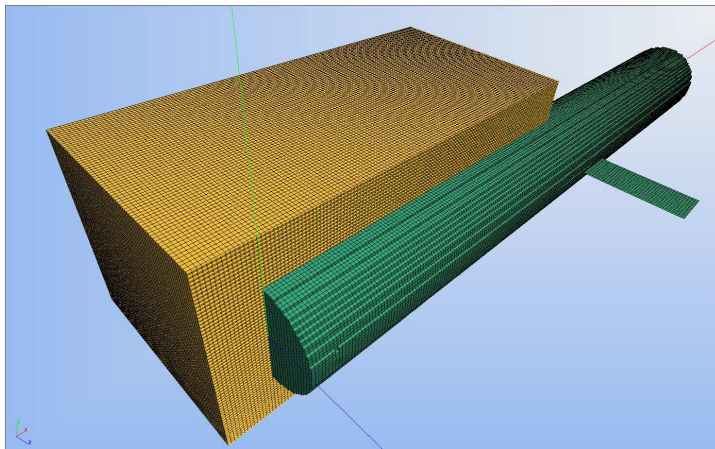


FIGURE 6.18 – Maillage traité par 4 processus au degré 2. La partie volumique représentée est associée à un processus. la surface du GENECE n'a subi qu'une faible déformation : la précision géométrique n'est pas la plus importante dans ce test de performance.

lation de degré 0 vraiment trop imprécise. Les caractéristiques des maillages utilisés pour les différents degrés et différents nombres de périphériques sont présentées dans le tableau 6.5.

Degré	1	2	3	4
Un processus	1728000	549024	226981	125000
Deux processus	3456000	1142974	453962	250000
Quatre processus	6912000	2190510	907924	500000

TABLE 6.5 – Nombre de mailles dans les différents cas de figure.

Afin de répartir les mailles entre les processus, nous procédons par découpage successif. Le maillage est découpé en son milieu suivant l'axe comprenant le plus grand nombre de mailles. Nous obtenons la répartition pour deux processus. Puis nous séparons à nouveau les mailles suivant le second axe comprenant le plus grand nombre de mailles. Puisque le maillage n'est pas cartésien, cela peut conduire à générer une zone partagée entre les processus plus grands que nécessaire. Toutefois, cette approche a donné des résultats satisfaisants dans cette configuration. Le maillage utilisé dans le cas d'une approximation d'ordre 2 et quatre processus est représenté sur la figure FIGURE 6.18.

Nous calculons alors le nombre d'opérations effectuées par seconde dans ces trois cas tests. Pour cela, nous comptons « à la main » le nombre d'opérations

flottantes effectuées par les différents *kernels* lors d'une itération pour l'ordre d'approximation et le nombre de mailles considérées. En divisant le résultat obtenu par le temps de calcul d'une itération nous obtenons la puissance de calcul exprimée en FLOPS (FLoating-point Operations Per Second). Nous obtenons alors l'accélération due à la parallélisation MPI sur n cartes graphiques en effectuant le rapport entre la puissance de calcul obtenue avec ces n cartes et celle obtenue avec une carte. Ces valeurs sont présentées dans le tableau 6.6.

	Nombre de GPU	1	2	4
Ordre 1	Temps de calcul par itération (s)	0.59	0.79	0.99
	Transfert par itération (%)	0	25	40
	Puissance (GFLOPS)	109	163	260
	Accélération	/	1.5	2.38
Ordre 2	Temps de calcul par itération (s)	0.28	0.47	0.90
	Transfert par itération (s)	0	45	62
	Puissance (GFLOPS)	212	263	264
	Accélération	/	1.26	1.29
Ordre 3	Temps de calcul par itération (s)	0.21	0.60	1.0
	Transfert par itération (s)	0	67	79
	Puissance (GFLOPS)	265	185	223
	Accélération	/	0.7	0.8
Ordre 4	Temps de calcul par itération (s)	0.28	0.80	1.34
	Transfert par itération (s)	0	0.52	79
	Puissance (GFLOPS)	214	150	179
	Accélération	/	0.7	0.8

TABLE 6.6 – Performances sur 1, 2 et 4 cartes graphiques.

L'accélération apportée par la parallélisation sur plusieurs périphériques n'atteint pas la valeur optimale. Dans les différents cas, le temps dédié aux communications MPI est important, jusqu'à 80% du temps de calcul. Cet important temps de transfert s'explique par le transfert de données volumiques pouvant représenter un volume important et par le temps de préparation de ces données. En effet, ces données sont transférées entre l'hôte et le périphérique de calcul et doivent être réordonnées d'une manière similaire par les deux processus communiquant afin d'effectuer un échange de données cohérentes.

Même si cette parallélisation n'offre pas des performances extraordinaires, elle permet de traiter des cas d'une taille ne permettant pas d'être traité sur une carte graphique. Nous présenterons au chapitre suivant une parallélisation MPI permettant de retrouver de bonnes performances.

Conclusion

Dans ce chapitre, nous avons présenté différents cas tests permettant de valider les résultats obtenus avec le code GD parallèle. Ces cas tests montrent l'efficacité de ce code obtenue en tirant parti de la parallélisation OpenCL des calculs. Nous avons également mis en évidence une limitation de la stratégie de parallélisation MPI. En effet, celle-ci conduit à des accélérations insatisfaisantes. Afin de remédier à ce problème, nous définirons dans le chapitre suivant une nouvelle stratégie de parallélisation dans laquelle nous améliorerons l'adaptation des *kernels* de calculs aux zones et où nous modifierons la procédure de communication MPI afin d'améliorer les performances de cette parallélisation.

Chapitre 7

Algorithme Galerkin Discontinu parallèle asynchrone

Les chapitres 5 et 6 décrivent la première implémentation de la méthode GD et les tests effectués avec celle-ci. Cette première version a permis de mettre en évidence différentes limitations et optimisations possibles de l'implémentation de la méthode. Premièrement, il apparait nécessaire d'effectuer une asynchronisation des calculs et transferts MPI afin d'obtenir une efficacité optimale. Deuxièmement, certaines fonctionnalités ne peuvent pas être implémentées en utilisant la structure de donnée actuelle. On peut citer l'utilisation d'un pas de temps local et la prise en compte de maillage non conforme. Et enfin, dans le cas de l'utilisation d'un ordre d'approximation variable, un gain important pourrait être obtenu en spécialisant les *kernels*.

Ces constatations nous ont conduits à la définition d'une nouvelle structure de données dans laquelle le concept de zones homogènes prend une place centrale. Les mailles sont comme précédemment réparties en zones homogènes, mais à présent chaque zone est munie d'un jeu de *kernels* spécialisés en fonction de l'approximation et de la nature des éléments. Cette structure permet alors de lever les limitations observées et d'améliorer les performances du code.

7.1 Structure de zones

Le maillage est découpé en zones formées d'éléments de même nature. Les éléments d'une zone ont la même géométrie et le même espace d'approximation. Nous pouvons ainsi appliquer à ces éléments un traitement similaire.

Ainsi, à partir des caractéristiques de la zone, nous construisons les *kernels* qui lui seront appliqués. Pour plus d'efficacité, chaque zone constituée de volumes effectuera des calculs mettant en jeu exclusivement des données portées par ces mêmes volumes. Ainsi, en plus de ces zones volumiques, nous devons définir des zones d'interfaces qui permettront la communication entre les zones volumiques. Ces zones peuvent également servir à effectuer un traitement spécifique aux données extraites des zones volumiques voisines. Cela permet par exemple de traiter un raccord non conforme entre zones.

7.1.1 Zone homogène

Une zone homogène est constituée d'éléments volumiques d'une même nature géométrique et munis de la même base d'approximation. Les *kernels* associés à une zone sont construits en utilisant les données géométriques et d'interpolation communes à ces éléments. Dans les *kernels*, différentes fonctions sont appelées afin de calculer les données utilisées lors du calcul. Ces fonctions sont définies uniquement par la nature de la zone et sont dédiées au calcul

- des points et poids d'intégration,
- de la transformation de l'élément de référence vers l'élément physique,
- de la transformation de l'élément physique vers l'élément de référence,
- de la comatrice de la transformation géométrique,
- de la valeur des fonctions de base,
- du gradient des fonctions de base.

Plusieurs ensembles de *kernels* ont été développés, des *kernels* pouvant être appliqués à tout type de maille et des *kernels* implémentant des optimisations applicables aux maillages hexaédriques. Les *kernels* génériques implémentent la méthode décrite en 5.1.3. Les *kernels* optimisés pour hexaèdres implémentent les optimisations décrites en 5.1.4.

Ainsi, la construction des *kernels* est effectuée en concaténant les sources des fonctions citées plus haut et des sources des *kernels* pouvant s'appliquer à la zone. Actuellement, nous avons mis en place la prise en charge des maillages hexaédrique et le développement d'autres éléments est en cours. Cette structure peut par exemple permettre la prise en charge de maillage tétraédrique en définissant les fonctions adéquates.

Cette approche permet également d'écrire « en dur » dans les sources des *kernels* certaines informations, nous évitant de les lire en mémoire. Par exemple, le degré d'approximation est stocké dans une macro, puisqu'un kernel différent est compilé pour chaque zone.

Les calculs sont alors effectués au moyen des différents *kernels* spécifiques. Toutefois, chaque zone ne peut calculer que les contributions provenant de mailles qu'elle contient. Afin d'appliquer les contributions surfaciques situées à l'interface entre deux zones, nous définissons entre chaque zone voisine une zone d'interface. La section suivante présente ce nouveau type de zones.

7.1.2 Zone d'interface

Nous avons décrit dans ce qui précède le traitement appliqué aux zones volumiques. Ces zones sont couplées par le moyen de zones d'interfaces. Ces interfaces peuvent être entre zones présentant des interpolations différentes ou sur des raccords non conformes entre mailles ou les deux à la fois. Actuellement, nous traitons bien sûr les raccords conformes. Nous avons aussi programmé le raccord non conforme entre une zone raffinée et une zone non raffinée dans un cas simple : chaque élément quadrangulaire de la zone non raffinée se trouve en vis-à-vis de quatre éléments quadrangulaires de la zone raffinée.

Nous commençons par définir une zone surfacique à l'interface entre les zones volumiques. Celle-ci est constituée des faces des éléments volumiques incluses dans l'interface. Les faces constituant l'interface sont orientées de manière à ce que leur normale pointe toujours vers la même zone volumique. Nous appellerons cette zone, la zone de droite et l'autre zone la zone de gauche. Dans le cas d'un raccord non conforme, nous construisons l'interface avec les faces provenant du côté le plus raffiné. Ces faces sont groupées de manière à ce qu'un groupe de faces soit voisin d'un unique élément non raffiné.

Les traitements appliqués aux interfaces sont effectués par une structure dédiée. Les *kernels* utilisés sont propres à chaque interface et sont construits en fonction de la situation, *i.e.* en fonctions des interpolations des zones volumiques et du type de raccord (conforme ou non). Quelle que soit la situation, les traitements donneront lieu aux mêmes types de tâches :

- extractions des champs : calcul des champs W_L et W_R sur l'interface en utilisant les données volumiques portées par les zones volumiques de gauche et droite,
- calcul des flux : calcul des flux sur l'interface en fonction des champs W_L et W_R ,
- application de la contribution surfacique : ajout des intégrales surfaciques (4.39 page 76) aux zones de gauche et droite.

Dans un premier temps, nous extrapolons les champs provenant de chaque côté de l'interface aux points d'intégration des faces constituant l'interface.

Pour cela, nous parallélisons les calculs sur les points d'intégration des faces. Ces calculs sont effectués par des *kernels* spécifiques construits en suivant la même stratégie que pour les zones volumiques, c'est-à-dire que les *kernels* sont choisis suivant leur adaptation à la géométrie présente et appellent des fonctions dont les sources correspondent à la nature des zones. Nous compilons un *kernel* pour chaque zone surfacique.

Dans un second temps, nous calculons et stockons en mémoire globale les flux aux points d'intégration. Cette phase est parallélisée sur les points d'intégration des faces. Comme précédemment un *kernel* spécialisé effectue cette opération.

Enfin, les termes surfaciques sont ajoutés dans chacune des zones voisines. Pour cela, nous utilisons deux *kernels* adaptés parallélisés sur les points surfaciques. Ces *kernels* utilisent tous deux la même formule de quadrature ajustée à l'ordre d'approximation le plus élevé et la géométrie la plus raffinée. Ainsi, les flux de part et d'autre de l'interface sont les mêmes, c'est-à-dire que le schéma reste conservatif.

Remarque 18. *Un intérêt de cette structure est la possibilité de modifier suivant le contexte le traitement des données. Par exemple, si la zone est à l'interface avec un autre sous-domaine les transferts de données MPI sont considérés comme un traitement spécifique appliqué à cette interface.*

7.1.3 Zone d'interface MPI

Comme dans la première implémentation, le volume de calcul peut être réparti en plusieurs sous-domaines traités chacun par un périphérique OpenCL. Sur chaque périphérique OpenCL, les calculs sont soumis par un unique processus MPI. Les processus traitant des zones voisines doivent alors échanger des données afin de pouvoir calculer la dérivée temporelle des champs. Ce transfert sera réalisé à partir d'une zone d'interface spécifique. Cette zone n'aura qu'une zone volumique voisine, la seconde fait partie d'un autre sous-domaine.

Cette zone est générée comme les zones d'interface présentées précédemment. Seules l'extrapolation des champs surfaciques et l'application du flux diffèrent. La phase d'extrapolation des champs est effectuée en deux temps. D'abord, nous extrapolons les champs provenant de la zone volumique de la même manière que pour une zone d'interface non MPI. Dans un second temps, une communication est engagée afin d'échanger les champs extrapolés avec le sous-domaine voisin. Dans le sous-domaine voisin, les mêmes étapes ont été suivies. Ainsi, une fois la communication terminée, les deux

sous-domaines ont connaissance des champs provenant des deux côtés de l'interface.

Les calculs des termes surfaciques sont alors effectués de la même manière que pour les interfaces entre zones homogènes. Enfin, nous appliquons les contributions surfaciques portées par l'interface. Ces contributions ne sont accumulées que dans la zone volumique appartenant au sous-domaine traité.

Cette stratégie présente principalement trois avantages comparés à la stratégie présentée dans la section 5.2 page 105

- le volume de données transférées est moins important, puisque nous transférons des données surfaciques plutôt que volumiques.
- les données échangées sont ordonnées de la même manière dans chaque processus ce qui diminue le temps de pré- et post-traitement,
- les traitements pouvant être appliqués aux zones d'interface non MPI s'appliquent naturellement aux zones d'interface MPI ; nous pouvons par exemple faire coïncider un raffinement de maillage avec une surface d'échange MPI.

Afin d'illustrer l'utilisation de cette structure de données, définissons un exemple composé de trois zones volumiques traitées par deux processus MPI. Un schéma représentant cet exemple est visible sur la figure FIGURE 7.1 page suivante. Le processus 1 traite le sous-domaine composé de la zone volumique 1 et des zones surfaciques 1 et 2. Le processus 2 traite le sous-domaine composé des zones volumiques 2 et 3 et des zones surfaciques 1', 2' et 3. Les communications MPI sont effectuées par les zones surfaciques 1, 1' et 2, 2'. Les raccords non-conformes sont traités par les zones surfaciques 1, 1' et 3. Dans cet exemple, les zones surfaciques 1, 1' effectuent à la fois des communications MPI et traitent un raccord non-conforme. Nous utiliserons cet exemple par la suite pour illustrer nos propos.

Conclusion :

Dans cette section, nous avons présenté la structure de données faite d'une répartition en zones homogènes des mailles, chacune étant traitée à l'aide de *kernels* adaptés à sa nature géométrique et son interpolation. Nous avons également décrit une stratégie de communication MPI permettant de limiter le coût des transferts. Nous allons maintenant nous intéresser à l'utilisation de cette structure de données lors d'un calcul. Nous présenterons l'algorithme global, puis la répartition en tâches des calculs et enfin l'asynchronisation de ces tâches.

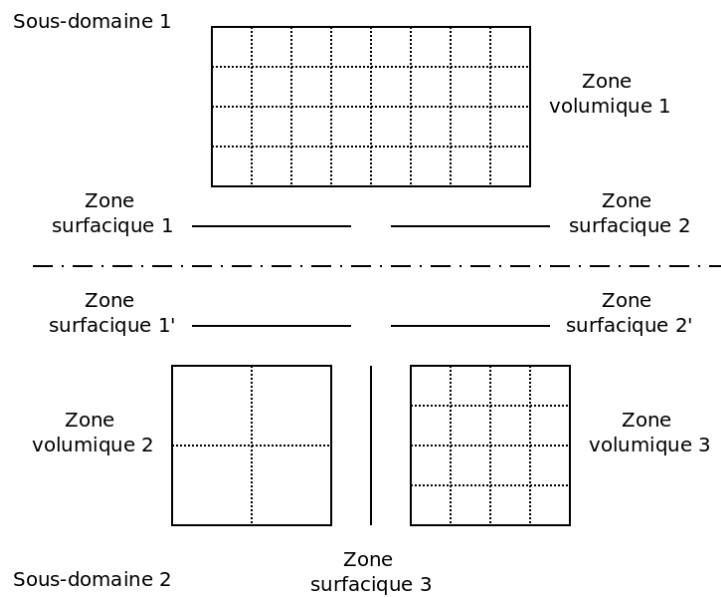


FIGURE 7.1 – Exemple de maillage traité par deux processus. Le processus 1 traite la zone volumique 1. Le processus 2 traite les zones volumiques 2 et 3. Les communications MPI sont effectuées par les zones surfaciques 1, 1' et 2, 2'. La zone surfacique 3 traite le raccord non conforme entre les zones volumiques 2 et 3. Les zones surfaciques 1 et 1' traitent également le raccord non-conforme entre les zones volumiques 1 et 2.

7.2 Déroulement des calculs

7.2.1 Un algorithme GD à deux niveaux

Le domaine de calcul est constitué de différentes zones volumiques séparées par des zones surfaciques. Dans un premier temps, un algorithme GD parallèle est appliqué dans chacune des zones volumiques. Ensuite, les termes surfaciques portés par les interfaces entre zones sont ajoutés.

L'algorithme GD est constitué de deux étapes : une étape volumique et une étape surfacique. L'algorithme global appliqué à l'ensemble des zones volumiques et surfaciques est également constitué d'une étape volumique (le calcul effectué par chaque zone volumique) et d'une étape surfacique (l'ajout des contributions des zones d'interfaces). Ainsi nous voyons se dessiner un algorithme GD à deux niveaux.

Nous avons un algorithme GD global appliqué à l'ensemble des zones volumiques et surfaciques et des algorithmes GD locaux appliqués aux éléments et faces des zones volumiques. Chaque GD local est indépendant, puisque les termes surfaciques portés par les interfaces entre zones sont traités dans des *kernels* différents. Les étapes de l'algorithme GD global peuvent être appliquées en parallèle, à condition de respecter certaines dépendances entre les calculs.

Nous allons donc maintenant expliquer comment nous avons parallélisé de façon asynchrone les opérations de l'algorithme GD global. Pour cela, nous commençons par définir les différentes tâches constituant l'avancée d'un pas de temps.

7.2.2 Graphe des tâches

L'avancée d'un pas de temps peut être décrite sous la forme d'un graphe de tâches. Chaque pas de temps de la méthode RK2 peut être vue comme deux pas de temps d'une méthode d'Euler ayant chacun un pas de temps différents. Ainsi, nous ne décrivons le graphe que d'un de ces demi-pas de temps.

Dans chacune des zones volumiques, les calculs sont divisés en deux tâches

- Calcul des intégrales volumiques et surfaciques,
- Application de la matrice de masse et avancée d'un demi-pas de temps.

Les traitements associés à une zone surfacique sont répartis en cinq tâches

- Extraction des champs provenant de la zone de gauche,
- Extraction des champs provenant de la zone de droite,
- Calcul des flux,
- Application des flux à gauche,
- Application des flux à droite.

Dans le cas d'une zone entre deux parties du maillage traitées par des processus différents, la tâche d'extraction des champs provenant de la zone de droite est une communication MPI. De même, l'application des flux à droite est une tâche vide.

Certaines tâches ne peuvent être exécutées qu'une fois d'autres terminées. Ainsi, nous pouvons construire un graphe des tâches représentant ces dépendances. Les figures FIGURE 7.2 page suivante et FIGURE 7.3 page 142 représentent de tels graphes de tâches. Ces graphes correspondent à l'exemple donné dans la section 7.1.3 page 136 comportant deux sous-domaines, l'un constitué d'une zone et l'autre de deux, et présentant un raccord non-conforme. Cet exemple est schématisé sur la figure FIGURE 7.1 page 138. Sur les figures FIGURE 7.2 page suivante et FIGURE 7.3 page 142, les tâches associées à des zones volumiques sont représentées par des rectangles bleus. Les tâches associées à des zones surfaciques ne faisant pas intervenir de communications MPI sont représentées par des ovales rouges. Les tâches de communications MPI sont représentées par des losanges rouges. Le nom d'une tâche est constitué du sous-domaine contenant la zone associée à cette tâche, du nom de cette zone et de l'intitulé de la tâche.

7.2.3 Asynchronisation des transferts

Les transferts de données via la bibliothèque MPI sont des opérations ne faisant intervenir que les ressources de l'hôte. OpenCL permet d'effectuer des calculs de manière asynchrone. Nous pouvons soumettre l'exécution d'une tâche au périphérique OpenCL et pendant son exécution effectuer d'autres opérations. Ainsi il est possible d'effectuer des calculs durant le temps passé à transférer des données entre le GPU et l'hôte ou pendant les communications MPI.

Pour cela, nous utiliserons le système d'évènements et de file d'attente fourni par OpenCL. Nous pouvons associer à l'exécution d'un *kernel* un évènement qui contient son statut d'exécution (en attente, commencée, terminée). De plus, le démarrage d'un *kernel* peut être rendu dépendant de la terminaison d'une liste d'évènements, c'est-à-dire que la soumission du *kernel* n'entraîne

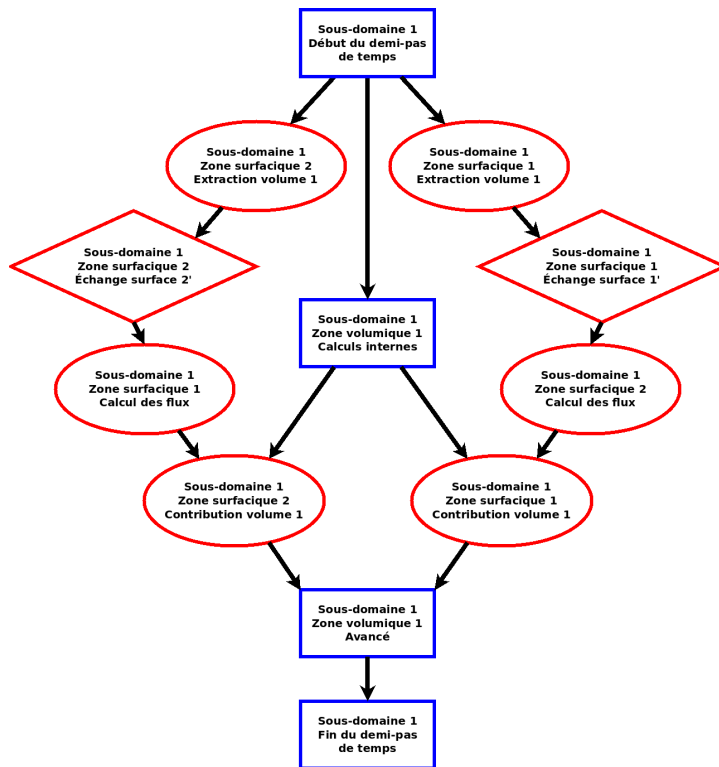


FIGURE 7.2 – Graphe des tâches associé au sous-domaine 1 représenté sur la figure FIGURE 7.1 page 138. Les rectangles bleus représentent les tâches associées à des zones volumiques. Les ovales et losanges rouges représentent les tâches associées à des zones surfaciques. Les tâches de communication MPI (losanges rouges) sont liées aux tâches MPI représentées sur la figure FIGURE 7.3 page suivante.

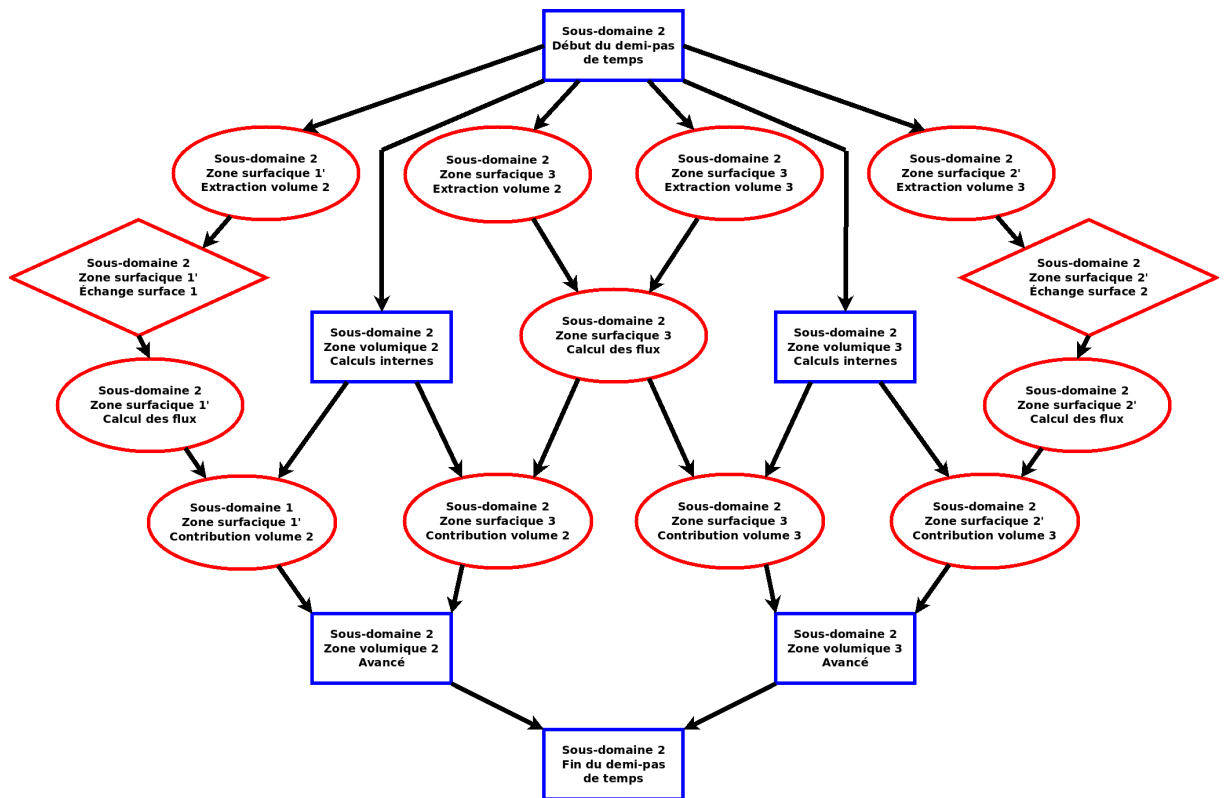


FIGURE 7.3 – Graphe des tâches associé au sous-domaine 2 représenté sur la figure FIGURE 7.1 page 138. Les rectangles bleus représentent les tâches associées à des zones volumiques. Les ovales et losanges rouges représentent les tâches associées à des zones surfaciques. Les tâches de communication MPI (losanges rouges) sont liées aux tâches MPI représentées sur la figure FIGURE 7.2 page précédente.

pas immédiatement son exécution, mais que celle-ci démarrera lorsque les évènements de la liste auront un statut « terminé ».

OpenCL permet également de définir des évènements dont le statut est modifié par l'utilisateur. Nous pouvons alors faire dépendre l'exécution d'un *kernel* d'une tâche exécutée par le périphérique hôte en lui associant un tel évènement. En associant un tel évènement à une tâche de communication MPI, celle-ci peut être incluse dans le graphe de dépendance des tâches géré au moyen des évènements OpenCL. Une tâche de communication MPI a alors un comportement similaire à une tâche OpenCL. Une tâche de communication est mise en file d'attente. Après cette soumission, le programme hôte continue son exécution. Le transfert de données ne commencera qu'une fois la tâche sortie de la file d'attente *i.e.* une fois que toutes les tâches dont elle dépend sont terminées.

Nous voulons de plus rendre dépendant le début de transfert de données de la terminaison de l'extraction des champs provenant de la zone volumique voisine. Néanmoins, la bibliothèque MPI ne permet pas de définir simplement ce type de dépendance. Pour cela, nous exécuterons les commandes de transfert dans des *threads*. Un *thread* est un processus exécuté en parallèle et partageant la mémoire du processus l'ayant lancé. Chaque processus (appelé *thread* principal) peut lancer plusieurs *threads* qui exécutent des instructions en parallèle. Durant l'exécution des *threads*, le *thread* principal continue lui aussi son exécution. Cette approche n'a à notre connaissance pas été exploitée jusqu'ici pour coupler un graphe des tâches OpenCL avec des appels MPI asynchrones.

Décrivons alors une tâche représentant une communication MPI. L'exécution de cette tâche dépend de la terminaison d'une liste L d'évènements et son statut est représenté par un évènement e . Cette tâche se déroule ainsi

- Soumission de transfert de données du GPU au CPU. Le départ de transfert dépend de la terminaison de tous les évènements de la liste L . Son statut est stocké dans un évènement $e_{GPU \rightarrow CPU}$.
- Le *thread* chargé des communications MPI est lancé. À ce *thread* sont donnés l'évènement $e_{GPU \rightarrow CPU}$ et un évènement e_{MPI} qui représentera son statut. Ce *thread* effectue alors les commandes suivantes
 - Une commande OpenCL permettant d'attendre la fin de l'évènement $e_{GPU \rightarrow CPU}$ est lancée.
 - Les communications MPI sont lancées.
 - Le *thread* attend la fin de la communication.
 - Le statut de l'évènement e_{MPI} est passé à "terminé".

Le lancement du *thread* étant non bloquant, l'exécution du *thread* principal continue durant l'exécution du *thread*.

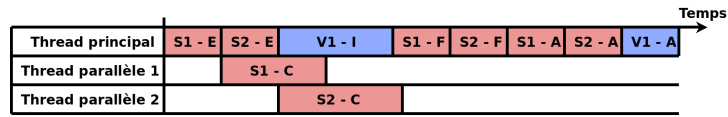


FIGURE 7.4 – Ordre d’exécution des tâches associées au sous-domaine 1 de l’exemple visible sur la figure FIGURE 7.1 page 138. Les tâches associées à la zone volumique sont représentées en bleu et nommées « V1 - I » (calculs internes) et « V1 - A » (Avancée en temps). Les tâches associées aux zones surfaciques sont représentées en rouge et nommées « S N° zone - » suivi d’une lettre représentant la tâche (E : extraction des champs, C : communication MPI, F : calcul des flux, A : accumulation de la contribution).

- Le transfert des données reçues du CPU au GPU est soumis. Le départ de ce transfert dépend de la terminaison de l’évènement e_{MPI} et son statut est stocké dans l’évènement e . La fin de ces transferts entraîne également la destruction du *thread* précédemment lancé.

En utilisant ces mécanismes, il nous est alors possible d’asynchroniser les calculs et transferts de données tout en garantissant la cohérence du résultat. De plus, cette approche empêche l’apparition de « *deadlock* », c’est-à-dire de blocage de l’exécution du programme, puisque les communications MPI sont toutes exécutées en parallèle par des *threads*.

Reprenons l’exemple dont le maillage est schématisé sur la figure FIGURE 7.1 page 138. Dans cet exemple, le maillage est réparti en deux sous-domaines. Le sous-domaine 1 contient une zone volumique et deux zones surfaciques. Les traitements associés à ce sous-domaine forment le graphe représenté sur la figure FIGURE 7.2 page 141. En utilisant la stratégie de communication asynchrone présentée dans cette section, les tâches pourront être exécutées par trois *threads* dans l’ordre représenté sur la figure FIGURE 7.4. Les exécutions des tâches MPI sont alors effectuées en parallèle. Le temps de transfert est entièrement recouvert par des calculs.

Conclusion

Dans ce chapitre, nous avons présenté une structure de données reposant sur la répartition des éléments en zones homogènes et sur la séparation des calculs en tâches. Cette structure permet de traiter chaque zone avec des *kernels* adaptés aux éléments qu’elle contient. Nous pouvons de plus prendre en charge des raccords non-conformes entre zones en appliquant un traitement spécifique sur ce type d’interfaces. Nous sommes également en mesure

d'effectuer certains calculs en parallèle des communications MPI grâce à une asynchronisation des tâches. Le prochain chapitre sera consacré d'une part à la validation du traitement des raccords non-conformes et d'autre part à l'évaluation de l'efficacité de l'asynchronisation des tâches.

Cette structure pourra également être étendue au traitement d'autres types de mailles (tétraèdres, prisme, *etc.*) et d'autres types d'interpolation (Gauss-Lobatto, sous-éléments, *etc.*), ainsi qu'à la simulation d'autres systèmes de lois de conservation.

Chapitre 8

Validations de l'algorithme asynchrone

Dans ce chapitre, nous présentons deux tests visant à valider la seconde implémentation de l'algorithme GD parallèle. Ces cas test ciblent chacun une fonctionnalité de cette implémentation, pour cela la scène considérée est très simple ; nous nous plaçons dans une configuration dans laquelle nous connaissons une solution exacte, nous permettant de valider simplement les calculs. Le premier cas test a pour but de valider la prise en compte des raffinements de maillage. Le second cas test permet de mettre en évidence l'apport de l'asynchronisation des calculs et transferts lors d'un calcul parallélisé avec MPI. Les différents cas tests sont effectués en simple précision.

8.1 Raffinement

8.1.1 Raffinement h

Ce cas test a pour objectif la validation de la prise en compte de maillage non conforme. Nous ne considérons que des raffinements où un hexaèdre est voisin par une de ces faces à quatre hexaèdres. Considérons le domaine de calcul $[0, 2] \times [0, 2] \times [0, 1]$. Nous injectons dans ce domaine une onde plane se propageant dans la direction donnée par le vecteur $V = (0.5, 0.5, \sqrt{2}/2)$ et polarisée de manière à ce que le champ électrique soit orienté suivant $(0.5, 0.5, -\sqrt{2}/2)$. Cette onde plane est monochromatique c'est-à-dire que

son amplitude est donnée par

$$g(t) = \sin(2\pi ft) \tag{8.1}$$

avec dans le cas présent $f = 300\text{MHz}$. Le domaine de calcul est composé de trois zones

- $[0, 1] \times [0, 1] \times [0, 1]$ composée de N^3 éléments,
- $[0, 1] \times [1, 2] \times [0, 1]$ composée de $(2N)^3$ éléments,
- $[1, 2] \times [0, 2] \times [0, 1]$ composée de $2 \times (2N)^3$ éléments.

Ces trois zones et le maillage obtenu pour un raffinement $N = 6$ sont visibles sur la figure FIGURE 8.1 page ci-contre.

Nous effectuons alors la simulation de cette configuration jusqu'à un temps égal à 2ns pour différents degrés d'approximation d et différents raffinements N . Nous calculons la norme L^2 de la différence entre les solutions numériques et la solution exacte. Pour un degré d'approximation fixé, le pas de temps est le même pour tous les raffinements et choisi afin que l'erreur due au schéma temporel soit petite. La composante E_x du champ électrique est représentée sur la figure FIGURE 8.1 page suivante pour $N = 6$ et $d = 4$.

Nous estimons alors l'ordre de convergence de la méthode aux différents ordres d'approximation. Les ordres de convergence obtenus sont reportés dans le tableau 8.1. Les courbes de convergence sont représentées sur la figure FIGURE 8.2 page 150. Nous obtenons des ordres de convergence correspondant aux ordres théoriques.

Degré	Ordre théorique	Ordre numérique
0	1	0.57
1	2	1.51
2	3	2.60
3	4	3.74
4	5	4.96

TABLE 8.1 – Ordres de convergence théoriques et numériques pour différents ordres d'approximation en présence d'un raffinement de maillage.

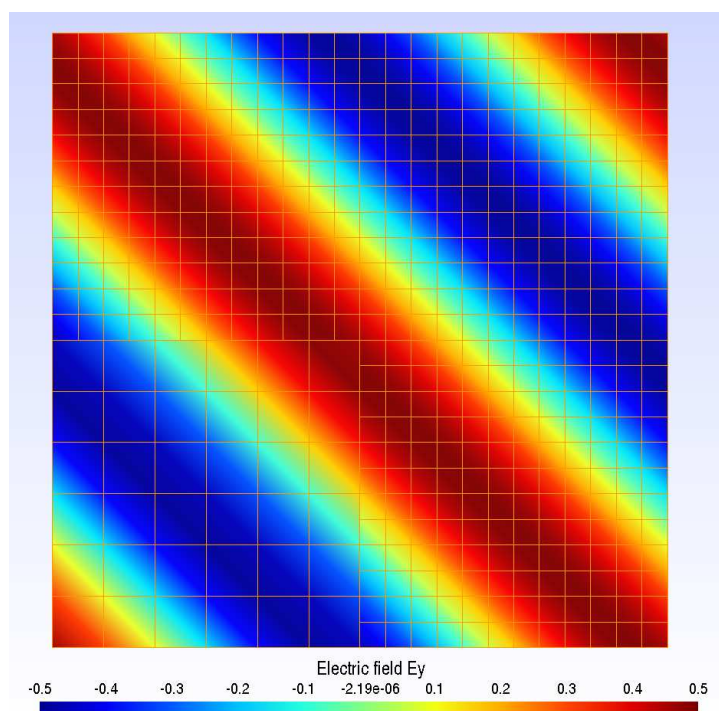


FIGURE 8.1 – Composante E_x du champ électrique calculé sur un maillage présentant un raffinement.

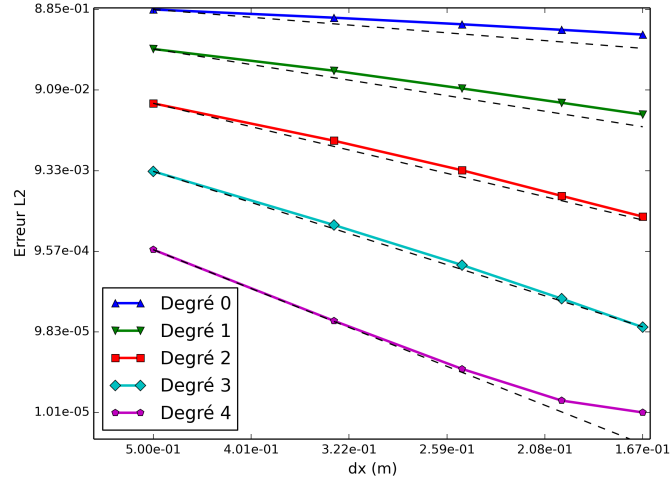


FIGURE 8.2 – Évolution de l’erreur en fonction du pas de discrétisation (échelle logarithmique) pour différents ordres d’approximation en présence d’un raffinement de maillage. Degré 0 : triangles bleus. Degré 1 : triangles inversés verts. Degré 2 : carrés rouge. Degré 3 : losanges cyan. Degré 4 : pentagone mauve.

8.1.2 Raffinement hp

Dans ce qui précède, nous avons utilisé un ordre d’approximation constant dans le maillage. Nous allons maintenant utiliser des ordres différents dans les zones suivant leurs raffinements. Nous choisissons les ordres d’approximation de façon à avoir la même densité de degrés de liberté dans chaque zone, *i.e.* en notant d_0 l’ordre de la zone non raffinée et d_1 l’ordre de la zone raffinée

$$(d_0 + 1)^3 = 8(d_1 + 1)^3 \quad (8.2)$$

puisque un élément non raffiné représente un volume égal à huit éléments raffinés. Nous prenons alors $d_0 = 3$ et $d_1 = 1$. Le maillage est constitué de deux zones cubiques de même volume et comme précédemment une onde plane monochromatique se propage dans le domaine. Nous calculons l’erreur commise en fin de simulation.

La figure FIGURE 8.3 page ci-contre représente en échelle logarithmique l’évolution de l’erreur en fonction du pas d’espace avec les ordres décrits plus haut (carrés verts), avec des degrés égaux à 1 (triangles rouges) et avec des degrés

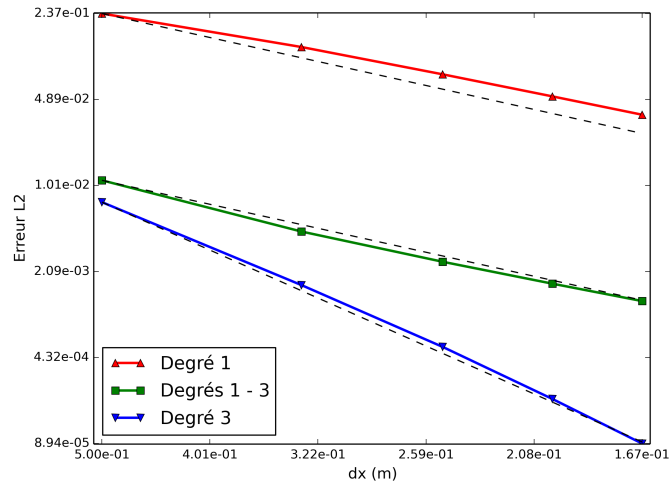


FIGURE 8.3 – Évolution de l’erreur en fonction du pas de discrétisation (échelle logarithmique). Degré constant = 1 (triangles rouges). Degré constant = 3 (triangles inversés bleus). Degré variable = 1,3 (carrés verts).

égaux à 3 (triangles inversés bleus).

Dans ces trois configurations, nous obtenons des ordres de convergence égaux à

- Degré 1 : 1.5,
- Degré 3 : 3.7,
- Degré 1-3 : 2.0.

Ainsi, lors de l’utilisation d’un ordre variable couplé à un raffinement de maillage, l’ordre de convergence de la méthode est donné par le plus faible ordre d’interpolation, comme nous pouvions nous y attendre. Toutefois, l’approximation est plus précise.

8.2 Scalabilité

Dans ce cas test, nous cherchons à estimer les performances de l’algorithme GD parallélisé sur plusieurs périphériques OpenCL. Nous allons utiliser la même approche que celle utilisée précédemment dans la section 6.5. Nous allons déterminer pour différents nombres de périphériques une puissance de calcul obtenue avec un maillage de taille maximale et en déduire l’accéléra-

tion due à la parallélisation.

8.2.1 Cas test

Comme précédemment, nous considérons un cas test simple pour lequel nous connaissons une solution exacte afin de pouvoir valider facilement les calculs. Le domaine de calcul est constitué de zones cubiques comprenant chacune 90^3 éléments. Chaque zone est traitée par une carte graphique. En terme de mémoire, ceci représente 3Go 200Mo par carte graphique. Nous propageons dans ce domaine une onde plane définie de la même manière que dans le cas test précédent. Nous effectuons ce test pour des ordres d'approximation allant de 0 à 4. Pour chaque degré d'approximation, le pas de temps utilisé dans chacun des tests est le même. Nous estimons alors la puissance de calcul par le rapport entre le nombre d'opérations nécessaires à une itération et le temps moyen de calcul d'une itération.

Nous nous intéressons ici à l'apport de l'asynchronisation des tâches de calcul et de transfert MPI. Nous considérons donc deux configurations du traitement des tâches. D'une part, les tâches sont exécutées de manière synchrone. Pour cela, les tâches sont rendues bloquantes en requérant la terminaison de chacune d'elle avant de passer à la suivante. D'autre part, les tâches sont exécutées de manière asynchrone. Dans ce cas, à chaque demi-pas de temps, les différentes tâches sont soumises et leur exécution gérée grâce au mécanisme d'événements fourni par OpenCL. Nous comparons également ces résultats à la puissance de calcul atteignable définie par le modèle « roofline » (voir [70]).

Nous effectuons ce cas test en parallélisant les calculs sur une, deux, quatre et huit cartes graphiques Nvidia Tesla K20m. Ce type de carte graphique possède une mémoire de 4Go et une puissance crête de 3.5TFlops.

8.2.2 Résultats

Dans les différents cas, nous calculons la norme L^2 entre la solution calculée et la solution exacte. Dans les différents cas de figure nous obtenons une erreur L^2 relative inférieure à 10^{-5} (suivant l'ordre d'approximation), nous permettant ainsi de valider les calculs.

Le tableau 8.2 représente les performances obtenues dans les différentes configurations. L'asynchronisation des calculs et transferts permet effectivement de diminuer le temps de calcul. Toutefois, ce gain n'excède pas le minimum

des temps de calcul et de transfert. Pour des calculs faits sur deux et quatre périphériques, le temps de transfert est presque négligeable (moins de 3%). Ce temps de transfert faible est dû d'une part au transfert de données surfaciques, ce qui représente $(d + 1)^2$ fois moins de données qu'un transfert volumique, d'autre part à la réduction des préparations des données aux transferts entre l'hôte et le périphérique de calcul. Le temps de transfert est toutefois important lorsque la parallélisation est effectuée sur huit périphériques (environ 30%). Cette augmentation soudaine du temps de transfert nous semble imputable à la complexification des communications et à l'augmentation du volume total de données transférées.

Nous obtenons alors une accélération proche de l'accélération théorique. Même dans le cas où les transferts ne sont pas asynchronisés, l'accélération est satisfaisante et bien supérieure à l'accélération obtenue avec la première stratégie de communication (donnée dans le tableau 6.6 page 130). Le gain dû à l'asynchronisation est intéressant lorsque nous utilisons huit périphériques de calcul, car le temps de transfert est alors important. Puisque les communications sont effectuées de manière asynchrone et ne requièrent aucun traitement des données, on pourrait s'attendre à atteindre l'accélération théorique. Or, l'ajout des contributions utilisant les données reçues ne peut pas être asynchronisé. En effet, ce traitement est effectué par un *kernel* qui ajoute ces contributions au tableau contenant la dérivée en temps des champs. C'est également dans ce tableau que sont ajoutées les contributions calculées par les zones volumiques. Afin de ne pas générer de conflits d'accès mémoire, ces traitements ne doivent pas être effectués en parallèle.

Examinons alors l'efficacité de l'asynchronisation de transferts dans le cas parallélisé sur huit périphériques. Pour cela, nous calculons la fraction du temps de transfert effectivement recouverte par un temps de calcul. Notons T_s et T_a les temps par itérations dans le cas synchrone et asynchrone. Supposons de plus que le temps de calcul, noté T_c , et de transfert, noté T_t , soient les mêmes dans les deux cas de figure. Définissons de plus, le « recouvrement » des calculs et transferts et notons le T_r . Les temps par itérations s'expriment alors

$$\begin{aligned} T_s &= T_c + T_t, \\ T_a &= T_c + T_t - T_r = T_s - T_r. \end{aligned} \tag{8.3}$$

La fraction de temps d'une itération effectivement recouverte est alors donnée

par

$$r = \frac{T_s - T_a}{T_s}. \quad (8.4)$$

En comparant cette valeur à la fraction de temps passé à effectuer les transferts de données T_i/T_s , nous obtenons une quantification de l'efficacité des recouvrements. Ces deux fractions de temps sont reportées dans le tableau 8.3. Aux ordres 1, 2 et 3, on obtient un recouvrement élevé. À l'ordre 4, le recouvrement est plus faible. Cela peut être dû à la complexification des communications.

Afin d'estimer l'efficacité de l'implémentation de cet algorithme, nous utilisons le modèle « roofline » qui permet de calculer la puissance de calcul maximale d'une implémentation. Ce modèle tient compte de deux caractéristiques du matériel utilisé : la puissance de calcul crête P et la bande passante B . Dans le cas présent, la puissance crête est de $P = 3.5\text{TFlops}$ et la bande passante est de $B = 208\text{Gbyte/s}$.

Ce modèle définit également l'intensité opérationnelle I d'une implémentation. Cette grandeur est donnée par le rapport entre le nombre d'opérations flottantes effectuées et la quantité de mémoire lue et écrite. Ces grandeurs permettent alors de calculer la puissance atteignable P_A défini par

$$P_A = \min(P, B \times I). \quad (8.5)$$

La puissance atteignable est alors bornée d'une part par la puissance crête du périphérique et d'autre part par la puissance maximale associée à l'implémentation. Cette puissance maximale est égale au produit de la bande passante B et de l'intensité opérationnelle I .

Nous pouvons alors tracer la puissance atteignable en fonction de l'intensité opérationnelle, voir la figure FIGURE 8.4 page ci-contre. Sur cette figure sont également représentées les intensités opérationnelles de la présente implémentation pour différents degrés d'interpolation (lignes verticales). Cette courbe nous indique que l'implémentation de l'algorithme GD est bien adaptée au matériel utilisé puisque nous ne sommes pas limités par la bande passante, sauf dans le cas $d = 1$. Ainsi, pour $d = 2, 3, 4$, la puissance atteignable est égale à la puissance crête, soit dans notre cas $P_A = 3.5\text{TFlops}$. Pour $d = 1$, la puissance atteignable vaut $P_A = 2.5\text{TFlops}$.

Nous obtenons une puissance de calcul se situant entre 25% et 30% de la

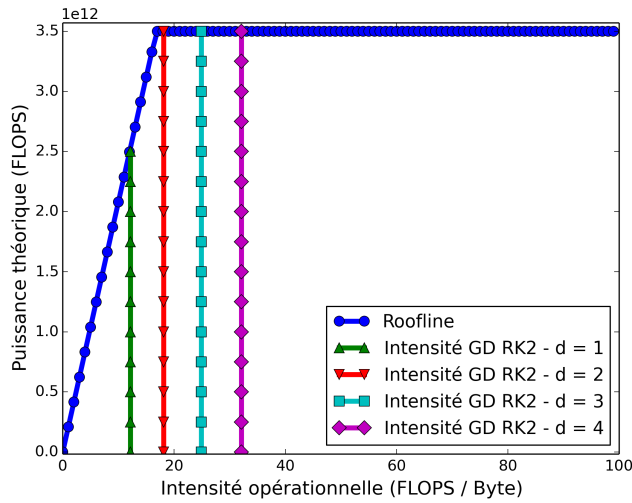


FIGURE 8.4 – Puissance atteignable donnée modèle « Roofline » de l’algorithme GD pour des degrés d’approximation allant de 1 à 4.

puissance maximale atteignable. Il est difficile de faire mieux, car certains accès mémoire non coalescents sont inévitables. Cette implémentation présente donc de très bonnes performances en tirant parti de la spécialisation des *kernels* de calcul, de la réduction de la quantité de données transférées et de l’asynchronisation des communications.

Conclusion

Dans ce chapitre, nous avons validé la seconde implémentation de l’algorithme parallèle GD. D’une part, nous avons étudié la validité des calculs lorsque le maillage présente une non-conformité avec des degrés d’approximation égaux et différents de par et d’autre de la non-conformité. Nous avons également mis en évidence l’apport de la stratégie de communication utilisant des transferts de données surfaciques et une asynchronisation des calculs et transferts. Cette parallélisation nous permet alors d’obtenir une accélération proche de l’accélération maximale.

		Nombre de GPU				
		1	2	4	8	
Ordre 1	Synchrone	Temps par itération (s)	0.525	0.535	0.552	0.80
		Transfert par itération (%)	0	<1	<1	33
		Puissance (TFlops)	0.255	0.50	0.968	1.34
		Accélération	/	1.96	3.80	5.25
	ASynchrone	Temps par itération (s)	0.524	0.532	0.543	0.55
		Puissance (TFlops)	0.255	0.502	0.984	1.94
Accélération		/	1.97	3.86	7.62	
Ordre 2	Synchrone	Temps par itération (s)	0.91	0.93	0.98	1.42
		Transfert par itération (%)	0	<1	1.5	35
		Puissance (TFlops)	0.602	1.18	2.23	3.08
		Accélération	/	1.96	3.71	5.13
	ASynchrone	Temps par itération (s)	0.913	0.93	0.951	0.97
		Puissance (TFlops)	0.6	1.18	2.30	4.70
Accélération		/	1.96	3.84	7.83	
Ordre 3	Synchrone	Temps par itération (s)	1.56	1.70	1.77	2.5
		Transfert par itération (%)	0	<1	2	32
		Puissance (TFlops)	1.01	1.84	3.53	5.07
		Accélération	0	1.83	3.53	5.01
	ASynchrone	Temps par itération (s)	1.56	1.61	1.67	1.72
		Puissance (TFlops)	1.01	1.96	3.78	7.34
Accélération		/	1.93	3.74	7.26	
Ordre 4	Synchrone	Temps par itération (s)	1.57	1.62	1.69	2.49
		Transfert par itération (%)	0	<1	2.5	30
		Puissance (TFlops)	1.20	2.32	4.45	6.04
		Accélération	/	1.94	3.72	5.04
	ASynchrone	Temps par itération (s)	1.57	1.62	1.69	2.19
		Puissance (TFlops)	1.20	2.32	4.45	6.87
Accélération		/	1.94	3.72	5.74	

TABLE 8.2 – Performance synchrone et asynchrone.

Ordre	1	2	3	4
Transfert (%)	33	35	32	30
Recouvrement (%)	31	32	31	12

TABLE 8.3 – Fraction du temps par itération due aux transferts de données et effectivement recouvert.

Chapitre 9

Modèles de fils

Nous avons présenté dans les chapitres précédents une méthode Galerkin Discontinue parallèle appliquée à l'électromagnétisme. Nous sommes alors en mesure de simuler diverses configurations au moyen de modèles physiques permettant de simuler des sources électromagnétiques ou des matériaux surfaciques et volumiques. Il est également important de pouvoir modéliser des structures filaires dans ce type de simulation. Différents modèles de fils ont été développés pour les schémas numériques FDTD (voir [35, 30]), éléments finis (voir [22]) ou volumes finis (voir [36]). Des travaux ont également été menés afin de définir un modèle de fil couplé à une méthode GD (voir [56, 72]). Lorsqu'un courant circule dans un fil, cela génère un second membre singulier, de type mesure de Dirac linéique, au second membre des équations de Maxwell. Cette singularité génère en général des oscillations de Gibbs dans les méthodes numériques. La méthode GD n'échappe pas à ce phénomène. Nous proposons dans ce chapitre, une méthode numérique pour supprimer la singularité des champs. Nous présentons la méthode sur un cas simplifié 1D. L'extension aux dimensions supérieures est en cours.

9.1 Principe des modèles de fil

Les différents modèles de fils que l'on peut trouver dans la littérature reposent sur un couplage des équations de Maxwell décrivant l'évolution des champs électromagnétiques dans l'espace avec un système de lois de conservation (9.1 page suivante) qui régit la propagation du courant I et de la charge q . Nous notons v la vitesse de propagation des ondes le long du fil, l l'abscisse

curviligne et \vec{l} le vecteur tangent unitaire le long du fil.

$$\begin{aligned}\partial_t I &= -v^2 \partial_l q \\ \partial_t q &= -\partial_l I\end{aligned}\tag{9.1}$$

Ce système est résolu à l'aide d'un schéma numérique 1D, en général de la même famille que le schéma 3D.

Le couplage des deux systèmes est effectué par l'introduction de termes sources. D'une part, une source dépendant du champ électrique tangent $E_t = E \cdot \vec{l}$ au fil est introduite dans la première équation de (9.1)

$$\partial_t I = -v^2 \partial_l q + \frac{E_t}{L}.\tag{9.2}$$

Cette source est modulée par une inductance L .

Le calcul de cette inductance est délicat. Par exemple, Holland dans [35] décrit une méthode pour calculer L pour des applications FDTD. Les arguments sont à la fois physiques et numériques. Il considère la maille de Yee comme un blindage entourant le fil, ce qui conduit à la formule

$$L = \frac{\mu_0}{2\pi} \left(\ln \frac{R}{a} - \frac{1}{2} + \frac{a^2}{2R^2} \right)\tag{9.3}$$

où a est le rayon du fil et R est le rayon du disque ayant une surface égale à la surface transverse de la cellule.

Cette formule peut être modifiée afin de s'adapter au couplage avec une méthode éléments finis (continus ou discontinus), dans [22] l'auteur propose une inductance égale à la moyenne

$$L = \frac{1}{M} \sum_{i=0}^M \frac{\mu_0}{2\pi} \ln \frac{R_i + a}{2a}\tag{9.4}$$

où R_i est la distance maximale du fil au $i^{\text{ème}}$ degré de liberté (dans notre cas les points d'intégration) qui sont au nombre de M . La méthode a été justifiée de manière plus mathématique dans [13].

Pour le couplage des champs sur le fil, il suffit de calculer le champ électrique tangent le long du fil grâce à l'interpolation GD. Ce calcul ne présente

pas de difficultés particulières. Mais, puisque le champ est singulier, il faut s'attendre à interpoler un champ dans une zone d'oscillations de Gibbs...

Le couplage inverse qui consiste en l'introduction d'une source de courant sur un support linéique dans les équations de Maxwell ne présente pas non plus de difficultés de principe : il suffit de calculer l'action de la mesure courant sur les fonctions test GD. Avec une source de courant, le schéma GD s'écrit

$$\begin{aligned}
\forall L \in \mathcal{M}, \forall \psi \in \mathcal{H}_h, \text{supp}(\psi) \subset \bar{L}, \\
\int_L \partial_t W \psi dx - \int_L F(W, W, \nabla \psi) dx + \\
\int_{\partial L \setminus \partial \Omega} F(W_L, W_R, n) \psi_L ds + \\
\int_{\partial \Omega} F_b(W_L, n) \psi_L ds = - \int_L S \psi_L dx.
\end{aligned} \tag{9.5}$$

Notons Γ le chemin du fil et δ_Γ la mesure de Dirac associé à Γ . Cette distribution est définie par

$$\langle \delta_\Gamma, \varphi \rangle = \int_\Gamma \varphi. \tag{9.6}$$

Avec ces notations, la source intervenant dans le schéma GD s'écrit

$$\int_L S \psi_L dx = \int_{l_0}^{l_1} \begin{pmatrix} I(t, l) \vec{l} \\ 0 \end{pmatrix} \psi_L(x(l)) dl \tag{9.7}$$

par définition de la masse de Dirac et en notant l_0 et l_1 les abscisses sur le fil tel que

$$l([l_0, l_1]) = \Gamma \cap L. \tag{9.8}$$

Il est donc en pratique assez facile de coupler un modèle de fil avec les équations de Maxwell. Cependant, les résultats sont en général décevants ou instables, à cause des oscillations de Gibbs. Ces oscillations apparaissent, car il est bien sûr impossible d'interpoler précisément des champs singuliers avec des polynômes.

Plusieurs solutions à ce problème ont été proposées. Par exemple dans [72], l’auteur étend la méthode développée par [22] au schéma GD. Celle-ci consiste en la répartition de la source dans un cylindre entourant le fil au moyen d’une fonction de pondération.

Dans la section suivante, nous présenterons une méthode permettant de tenir compte de la singularité générée par le fil en séparant les champs en une partie prise en charge par la méthode GD et une partie singulière prise en charge par un calcul semi-analytique. Cette méthode est sans doute compliquée à mettre en oeuvre en 3D. Nous nous contentons dans ce travail d’exposer les idées sur un modèle 1D simplifié. L’extension au 3D est sans doute possible et en cours d’étude.

9.2 Prise en compte de la singularité

Nous présentons dans cette section une méthode permettant d’appliquer une source de courant dans le volume de calcul. Cette source permet de modéliser la présence d’un fil parcouru par un courant dans le volume de calcul. Le rayonnement engendré par une telle structure présente une singularité au voisinage du fil et ne peut donc pas être correctement représenté dans la base d’approximation utilisée dans le schéma GD. Nous traitons donc séparément les deux champs présents. D’une part les champs électromagnétiques ambiants qui appartiennent à l’espace d’approximation. D’autre part, un champ singulier traité séparément.

9.2.1 Modèle unidimensionnel

Commençons par décrire le modèle 1D dans lequel nous ajouterons une source ponctuelle de courant. Nous déduisons des équations de Maxwell tridimensionnelles (1.1 page 8) une formulation 1D en considérant des solutions qui ne dépendent que d’une variable d’espace, ici la coordonnée suivant l’axe x . L’équation (1.1 page 8) peut alors être écrite sous la forme d’une équation 1D liant les composantes non nulles des champs.

$$\partial_t E_y(x, t) + \partial_x H_z(x, t) = 0 \quad (9.9)$$

$$\partial_t H_z(x, t) + \partial_x E_y(x, t) = 0 \quad (9.10)$$

Dans la suite, nous écrirons ces équations sous la forme

$$\begin{cases} \partial_t E + \partial_x H &= 0 \\ \partial_t H + \partial_x E &= 0 \end{cases} \quad (9.11)$$

Les variables E et H étant scalaires.

On remarque que la condition de divergence nulle des champs est automatiquement vérifiée.

L'équation (9.11) peut être mise sous la forme d'un système de Friedrichs similaire à (2.3 page 28)

$$\partial_t W + A \partial_x W = 0, \text{ avec } W = \begin{pmatrix} E \\ H \end{pmatrix} \text{ et } A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \quad (9.12)$$

En suivant la section 2.2, nous déduisons de ce système d'équations une formulation GD. Nous considérerons un domaine de calcul formé d'un segment $\Omega = [a, b]$ discrétisé en un ensemble de n segments $L_i = [a_i, a_{i+1}]$ avec $a = a_0 < a_1 < \dots < a_n = b$. Nous ajoutons au système (9.12) des conditions limites définies par des matrices M_a et M_b . Ces matrices seront déduites des formulations des conditions aux limites de la même façon que dans le cas 3D.

Nous prenons comme espace des solutions l'espace de fonctions \mathcal{H}^2

$$\mathcal{H} = \{f \in L^2(\Omega), \forall i, f|_{L_i} \in C^1(\overline{L_i})\}. \quad (9.13)$$

La solution appartient alors à l'espace $C^1([0, T], \mathcal{H}^2)$. La formulation GD du problème s'écrit comme suit, avec M la matrice correspondant à la condition limite appliquée au point considéré.

Trouver \mathcal{W} appartenant à $C^1([0, T], \mathcal{H}^2)$, tel que

$$\begin{aligned}
\forall 0 \leq i \leq n, \forall \varphi \in \mathcal{H}, \text{supp}(\varphi) \subset \overline{L}_i, \\
\int_{a_i}^{a_{i+1}} \partial_t \mathcal{W} \varphi dx - \int_{a_i}^{a_{i+1}} F(\mathcal{W}, \mathcal{W}, \nabla \varphi) dx + \\
(1 - \delta_{i+1, n}) F(\mathcal{W}_i(a_{i+1}), \mathcal{W}_{i+1}(a_{i+1}), 1) \varphi_i(a_{i+1}) - \\
(1 - \delta_{i, 0}) F(\mathcal{W}_i(a_i), \mathcal{W}_{i-1}(a_i), -1) \varphi_i(a_i) + \\
\delta_{i+1, n} (A + M_b) \mathcal{W}_i(b) \varphi_i(b) + \delta_{i, 0} (-A + M_a) \mathcal{W}_i(a) \varphi_i(a) = 0.
\end{aligned} \tag{9.14}$$

Nous choisissons ici un flux numérique F décentré

$$F(W_L, W_R, n) = An^+ W_L + An^- W_R \tag{9.15}$$

(ici, $n = \pm 1$). Nous discrétisons alors l'espace des solutions en approximant dans chaque intervalle L_i la solution par une combinaison linéaire de polynômes de Lagrange associés aux points d'intégration de Gauss-Legendre sur cet intervalle.

9.2.2 Source ponctuelle de courant

Nous imposons une source de courant ponctuel dans le domaine de calcul situé en $x_0 = 0$. Cette source peut être mise sous la forme d'une masse de Dirac en x_0 multipliée par la valeur du courant j qui dépend du temps. Avec cette source, le système (9.11 page précédente) devient

$$\begin{cases} \partial_t E + \partial_x H &= -j \delta_{x_0} \\ \partial_t H + \partial_x E &= 0 \end{cases} \tag{9.16}$$

En appliquant une transformée de Laplace en temps, nous obtenons un système d'équations vérifié par les transformées de Laplace des champs \tilde{E} et \tilde{H} .

$$\begin{cases} p \tilde{E} + \partial_x \tilde{H} &= -\tilde{j} \delta_{x_0} \\ p \tilde{H} + \partial_x \tilde{E} &= 0 \end{cases} \tag{9.17}$$

qui peut être mis sous la forme d'un produit de convolution

$$\begin{pmatrix} p\delta & \delta' \\ \delta' & p\delta \end{pmatrix} \star \begin{pmatrix} \tilde{E} \\ \tilde{H} \end{pmatrix} = \begin{pmatrix} -\tilde{j}\delta_{x_0} \\ 0 \end{pmatrix} \quad (9.18)$$

Par un calcul symbolique, nous pouvons inverser la matrice intervenant dans l'équation précédente et ainsi obtenir une formulation de la solution recherchée. Pour cela, nous allons calculer l'inverse pour le produit de convolution du déterminant de cette matrice $p^2\delta - \delta' \star \delta'$. L'inverse de cette expression est la solution fondamentale de l'équation de Helmholtz G définie par

$$(p^2\delta - \delta' \star \delta') \star G = \delta. \quad (9.19)$$

Cette fonction a pour valeur $G(x) = \frac{1}{2p}\exp(-p|x|)$. La solution de l'équation est alors donnée par

$$\begin{aligned} \begin{pmatrix} \tilde{E} \\ \tilde{H} \end{pmatrix} &= G \star \begin{pmatrix} p\delta & -\delta' \\ -\delta' & p\delta \end{pmatrix} \begin{pmatrix} -\tilde{j}\delta_{x_0} \\ 0 \end{pmatrix} \\ &= \frac{-\tilde{j}}{2}\exp(-p|x - x_0|) \begin{pmatrix} 1 \\ \text{sgn}(x - x_0) \end{pmatrix}. \end{aligned} \quad (9.20)$$

Nous en déduisons l'expression des champs électromagnétiques en fonction du temps en appliquant la transformation de Laplace inverse. Dans ce cas, le terme $\exp(-p|x - x_0|)$ correspond à un retard de $|x - x_0|$. En supposant que la solution initiale est nulle et que les conditions aux limites sont des conditions transparentes, la solution vaut

$$\begin{cases} E(t, x) &= \frac{-1}{2}j(t - |x - x_0|) \\ H(t, x) &= \frac{-1}{2}\text{sign}(x - x_0)j(t - |x - x_0|) \end{cases} \quad (9.21)$$

Cette solution présente une singularité au point x_0 . Le champ magnétique et la dérivée spatiale du champ électrique sont tous deux discontinus en ce point. Ainsi, l'approximation de ces champs présentera des oscillations au niveau de cette discontinuité et le taux de convergence de la méthode en sera impactée.

9.2.3 Application directe de la source

Une première approche pour prendre en compte une source de courant ponctuelle dans le schéma GD (9.14 page 162) est d'appliquer sa définition en tant que masse de Dirac (9.7 page 159).

Dans le cas unidimensionnel, cela conduit à introduire dans le schéma GD une source sous la forme

$$\int_{a_i}^{a_{i+1}} S\psi_L dx = \begin{pmatrix} I(t) \\ 0 \end{pmatrix} \varphi_i(x_0). \quad (9.22)$$

Comme dans le cas 3D, cette source induira l'apparition d'oscillations dans la solution numérique et une chute de l'ordre de convergence de la méthode.

9.2.4 Approximation de Padé

La solution exacte des champs rayonnés par le fil est exprimée par une formule de potentiel retardé (9.21 page précédente) qui n'est pas locale en temps. De plus, cette solution n'est plus valable dans le cas d'une géométrie compliquée ou en présence de diélectriques non homogènes.

Dans un premier temps, nous nous affranchissons de cette non-localité en temps en remplaçant les champs rayonnés par une approximation. Pour cela, considérons l'approximation de Padé à l'ordre 1 du terme de retard $\exp(-p|x - x_0|)$.

$$\exp(-p|x - x_0|) \simeq \frac{1}{1 + p|x - x_0|} \quad (9.23)$$

Nous obtenons alors une approximation de la transformée de Laplace des champs singuliers. En notant \tilde{W}_S cette approximation, nous avons

$$\tilde{W}_S = \frac{-1}{2} \frac{1}{1 + p|x - x_0|} \tilde{j} \begin{pmatrix} 1 \\ \text{sign}(x - x_0) \end{pmatrix}. \quad (9.24)$$

Nous appellerons ce champ le champ singulier. En appliquant la transformation de Laplace inverse, nous obtenons une équation différentielle ordinaire

vérifiée par W_S (9.25). Cette équation annexe est couplée aux équations de Maxwell par l'intermédiaire d'un terme source.

$$|x - x_0| \partial_t W_S + W_S = \frac{-1}{2} j \begin{pmatrix} 1 \\ \text{sign}(x - x_0) \end{pmatrix} \quad (9.25)$$

Cette solution vérifie approximativement les équations de Maxwell

$$\partial_t W_S + A \partial_x W_S = \begin{pmatrix} j \delta_{x_0} \\ 0 \end{pmatrix} + \text{erreur}. \quad (9.26)$$

L'erreur commise est due à l'approximation de Padé et peut être diminuée en utilisant une approximation d'ordre plus élevée, mais entraînant une complexification des équations annexes vérifiées par W_S .

Nous allons maintenant séparer le champ total recherché W en la somme d'un champ portant la singularité W_S et d'un champ régulier W_d . Pour cela, définissons une fonction ϕ continument différentiable, dont le support est localisé autour de x_0 et telle que $\phi = 1$ sur un voisinage de x_0 . Posons alors, $W = W_d + \phi W_S$.

En injectant cette expression dans l'équation (9.12 page 161) avec pour second membre le terme source $J = (j \delta_{x_0}, 0)^T$, nous obtenons

$$\partial_t W_d + A \partial_x W_d + \phi (\partial_t W_S + A \partial_x W_S) = J - \partial_x \phi A W_S. \quad (9.27)$$

Puisque W_S vérifie (9.26) et que ϕ vaut 1 au voisinage de x_0 , nous obtenons ainsi l'équation vérifiée par W_d avec un terme source dépendant de W_S

$$\partial_t W_d + A \partial_x W_d = -\partial_x \phi A W_S. \quad (9.28)$$

Le terme source assure le couplage entre les champs rayonnés par le fil et les champs ambiants. La singularité du champ rayonné est située en x_0 , or ϕ est constante sur un voisinage de x_0 . Le terme source $-\partial_x \phi A W_S$ est donc régulier. Le champ W_d pourra donc être approximé correctement en utilisant le schéma GD. Le champ W_S est quant à lui calculé en résolvant l'équation différentielle ordinaire (9.25) au moyen d'un schéma RK2. Nous devons connaître W_s aux points d'interpolation de la solution W_d afin de reconstruire la solution $W = W_d + \phi W_S$. Ainsi nous calculerons les valeurs de W_S en ces points.

Nous allons maintenant comparer cette méthode avec la méthode « naïve » qui consiste à représenter la source de courant en utilisant la définition de la masse de Dirac décrite dans la section 9.2.3 page 164.

9.2.5 Comparaison des résultats

Dans cette section, nous comparons les résultats obtenus en utilisant la méthode présentée précédemment avec ceux obtenus en appliquant la source de courant en utilisant la définition de la masse de Dirac. Nous comparons les taux de convergence obtenus dans les différents cas avec ceux obtenus sans la présence d'une source de courant.

Nous considérons le segment $\Omega = [-1, 1]$, nous plaçons en 0 une source de courant dont la forme d'onde est donnée par une fonction j

$$j(t) = \exp\left(-\left(\frac{t-t_0}{\tau}\right)\right) \quad (9.29)$$

avec $t_0 = 1$ et $\tau = 0.4$. En $t = 0$, les champs sont nuls.

Nous discrétisons Ω en un nombre de segments impair afin que la source de courant soit placée au centre d'une maille.

Nous effectuons les simulations avec différents nombres de mailles et différents ordres d'approximation afin de calculer un ordre de convergence de la méthode. Les figures FIGURE 9.1 page 168 et FIGURE 9.2 page 168 représentent les champs électrique et magnétique exacts et calculés en utilisant les deux méthodes. Nous pouvons observer sur ces figures la présence d'une oscillation au niveau de la source de courant lorsqu'elle est appliquée au moyen d'une masse de Dirac. Cette oscillation disparaît lorsque nous utilisons la méthode prenant en compte la singularité. Pour l'instant, nous n'avons pas pris en compte l'action des champs sur le fil. Il est clair que cette prise en compte serait délicate si les champs présentent des oscillations près du fil.

Les ordres de convergence obtenus avec les différents modèles de source sont donnés dans le tableau 9.1. Ce tableau comprend également l'ordre de convergence de la méthode 1D sans source de courant. Dans ce cas, nous considérons la solution exacte donnée par

$$\begin{aligned} E(x, t) &= \cos(t - x), \\ H(x, t) &= \cos(t - x). \end{aligned} \quad (9.30)$$

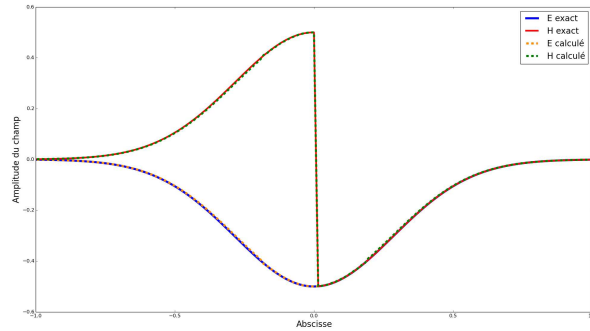


FIGURE 9.1 – Champs électrique et magnétique avec prise en compte de la singularité.

Les champs sont initialisés avec ces solutions et celle-ci est appliquée sur les bords du domaine au moyen de conditions aux limites de Silver-Müller inhomogènes. On peut remarquer que cet ordre est supérieur à l'ordre théorique de convergence dû à un phénomène de superconvergence. Ces résultats montrent que l'utilisation de la méthode présentée ici permet d'augmenter l'ordre de convergence de la méthode, même si l'ordre de la méthode seule n'est pas rattrapé. De plus, l'erreur commise est plus faible lorsque la singularité est prise en compte. Les erreurs L^2 obtenues avec un maillage comportant 45 mailles sont reportées dans le tableau 9.2.

Ordre	1	2	3	4
Sans source	2.0	4.2	6.3	8.5
Source Dirac	0.3	1.0	1.0	1.1
Prise en compte de la singularité	4.0	4.0	3.9	3.9

TABLE 9.1 – Ordre de convergence des différentes méthodes.

Ordre	1	2	3	4
Source Dirac	6.3×10^{-3}	1.7×10^{-3}	2.1×10^{-3}	5.0×10^{-4}
Prise en compte de la singularité	3.9×10^{-4}	2.6×10^{-5}	3.5×10^{-6}	7.6×10^{-7}

TABLE 9.2 – Erreur L^2 commise avec et sans prise en compte de la singularité.

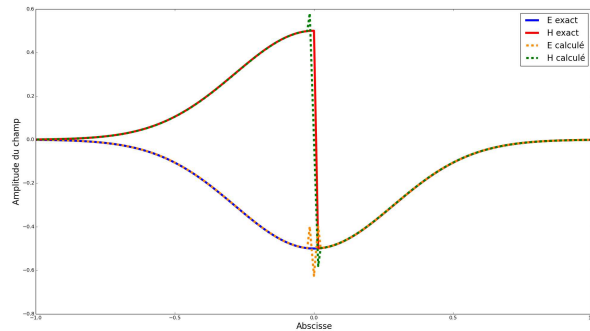


FIGURE 9.2 – Champs électrique et magnétique sans prise en compte de la singularité.

Conclusion

En conclusion, nous avons introduit dans ce chapitre la problématique des modèles de fil dans le schéma GD. Le rayonnement d'une source de courant portée par un fil mince peut entraîner l'apparition d'oscillations dans la solution et dégrader la précision des résultats. Nous avons alors présenté une méthode permettant d'introduire une telle source en conservant une approximation d'ordre élevé. Dans ce modèle, nous séparons les champs électromagnétiques en une partie régulière prise en charge par le modèle GD et une partie singulière vérifiant une équation annexe. Nous avons alors mis en œuvre cette méthode dans un schéma GD unidimensionnel. Dans ce cas, sans prise en charge de la singularité, l'ordre du schéma est dégradé et des oscillations de Gibbs apparaissent au niveau de la singularité. En appliquant la stratégie décrite dans ce chapitre, nous retrouvons un schéma d'ordre élevé et les oscillations sont supprimées.

Par la suite, ce modèle pourra être étendu au cas tridimensionnel. Pour cela, nous devons établir une équation vérifiée par les champs singuliers au voisinage du fil. Afin d'obtenir un modèle efficace, ces champs devront dépendre que des parties de fils proches. Nous espérons ainsi obtenir un modèle d'une bonne précision et bien adapté à une parallélisation.

Chapitre 10

Calcul de matrice de capacitance et inductance de section de fil

Une application utile dans le domaine de la compatibilité électromagnétique consiste en la simulation d'un réseau de câbles complexes dans leur environnement. Les champs environnants sont traités par une méthode 3D et couplés aux courants circulant sur les câbles. La propagation des courants et tensions sur le câble est simulée par une méthode 1D. Les interactions des différents conducteurs présents dans un câble sont représentées par des matrices de résistance, inductance, capacitance et conductance. Ces matrices dépendent de la géométrie de la section du câble et doivent être déterminées numériquement dans la plupart des cas. Les concepts abordés ici sont présentés plus en détail dans [55, 14]. Dans ce chapitre, nous nous intéresserons au calcul des matrices d'inductance et de capacitance. Pour cela, nous utilisons une méthode intégrale développée par B.L. Michielsen (ONERA). Les calculs nécessaires à l'obtention de ces matrices pour des réseaux de câbles complexes peuvent être longs. Nous allons donc paralléliser cette méthode en utilisant la bibliothèque OpenCL. Ces travaux ont été réalisés durant les premiers mois de cette thèse et ont permis la prise en main de la bibliothèque OpenCL.



FIGURE 10.1 – Exemple de section de câble.

10.1 Problème physique

Considérons un câble constitué d'un nombre arbitraire $N + 1$ de conducteurs et de gaines diélectriques. Pour une telle structure, nous souhaitons calculer les tensions U et courants I propagés sur chacun des conducteurs du réseau. Pour un câble infini orienté suivant l'axe z , I et U associé à un temps t et une abscisse z sur le câble les vecteurs $I(t, z) \in \mathbb{R}^{N+1}$ et $U(t, z) \in \mathbb{R}^{N+1}$. Sur un câble donné, ces grandeurs physiques vérifient l'équation des lignes de transmission

$$\begin{aligned} L\partial_t I + RI &= \partial_z U \\ C\partial_t U + GU &= \partial_z I \end{aligned} \tag{10.1}$$

Les matrices de résistance R , d'inductance L , de capacitance C et de conductance G décrivent l'interaction entre les différents conducteurs du câble. Ces matrices doivent alors être précalculées en utilisant un outil tiers. Dans seulement quelques cas très simples, ces matrices sont connues analytiquement.

Considérons une section de câble contenant un nombre $N + 1$ de conducteurs parfaits D_i baignant dans un milieu diélectrique de permittivité ε et pouvant être entourés d'une couche de diélectrique. Un exemple de section de câble est présenté sur la figure FIGURE 10.1. On munit les bords des conducteurs et les discontinuités de diélectrique d'une normale n orientée vers l'intérieur comme représenté sur la figure FIGURE 10.2 page suivante.

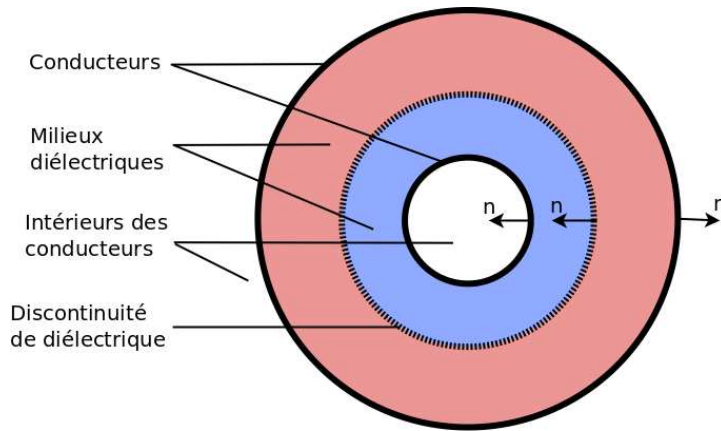


FIGURE 10.2 – Section contenant deux conducteurs (lignes pleines) et une discontinuité de diélectrique (ligne pointillée) munis de normales entrantes. Le contour du conducteur extérieur forme la limite du domaine.

Supposons alors que sur cette section de câble le champ électrique \mathcal{E} dérive d'un potentiel V , *i.e.* $\mathcal{E} = -\nabla V$. En injectant le potentiel dans l'équation de Maxwell-Gauss (1.2a page 8), nous obtenons l'équation de Laplace vérifiée par le potentiel électrique

$$-\nabla \cdot (\varepsilon \nabla V) = 0. \quad (10.2)$$

On remarque alors que sur les discontinuités de permittivité, le potentiel vérifie une condition de saut de la forme

$$[n \cdot \varepsilon \nabla V] = 0. \quad (10.3)$$

De plus, nous pouvons supposer que le potentiel électrique V est égal à une constante V_k sur chaque conducteur D_k , puisque dans les conducteurs le champ électrique est nul. La charge q_k du conducteur D_k est définie par l'intégrale

$$q_k = \int_{\partial D_k} n \cdot (\varepsilon \nabla \cdot V). \quad (10.4)$$

Définition 4. En prenant le conducteur 0 comme référence de potentiel, la matrice de capacité $C \in \mathcal{M}_N(\mathbb{R})$ lie les charges des conducteurs à leurs

potentiels.

$$\begin{pmatrix} q_1 \\ \vdots \\ q_N \end{pmatrix} = C \begin{pmatrix} V_1 \\ \vdots \\ V_N \end{pmatrix}. \quad (10.5)$$

La matrice d'inductance L peut être calculée de manière similaire. En effet, supposons que le champ magnétique \mathcal{H} dérive d'un potentiel vecteur A , $\mathcal{H} = \nabla \times A$. Sur la section considérée, le champ magnétique est tangent sur le plan de la section, ainsi seule la composante orthogonale du potentiel est non nulle, $A = (0, 0, \varphi)$. Le champ scalaire φ est appelé potentiel magnétique ou encore flux magnétique. En injectant le potentiel magnétique dans l'équation de Maxwell-Thomson (1.2b page 8), il vient

$$-\Delta\varphi = 0. \quad (10.6)$$

Nous pouvons, dans ce cas, également supposer que le potentiel magnétique φ est égal à une constante φ_k sur le conducteur D_k . L'intensité du courant I_k sur le conducteur D_k est donnée par

$$I_k = \int_{\partial D_k} n \cdot \nabla \cdot \varphi. \quad (10.7)$$

Définition 5. *En prenant le conducteur 0 comme référence de potentiel, la matrice d'inductance $L \in \mathcal{M}_N(\mathbb{R})$ lie les courants des conducteurs à leurs potentiels magnétiques.*

$$\begin{pmatrix} \varphi_1 \\ \vdots \\ \varphi_N \end{pmatrix} = L \begin{pmatrix} I_1 \\ \vdots \\ I_N \end{pmatrix}. \quad (10.8)$$

Nous remarquons alors que les matrices de capacitance et inductance lient des grandeurs physiques vérifiant toutes des équations de Laplace. Ainsi, en calculant la matrice de capacitance C_0 d'une section de câble en prenant une permittivité identiquement égale à la permittivité du vide, nous obtenons une matrice dont l'inverse est la matrice d'inductance de la section, à multiplication par une constante près

$$L = \frac{1}{c^2} C_0^{-1}. \quad (10.9)$$

La matrice d'inductance peut alors être calculée par le même procédé que la matrice de capacité en ne considérant pas les discontinuités de diélectriques.

10.2 Schéma numérique

10.2.1 Équation intégrale

Dans cette section, nous déduirons de l'équation de Laplace vérifiée par le potentiel électrique V une équation intégrale liant V aux charges électriques, pour cela nous nous baserons sur des travaux effectués à l'ONERA. Des travaux similaires sont présentés dans [64, 55].

Commençons par donner les notations que nous utiliserons dans la suite de ce chapitre.

Soit $\Omega \subset \mathbb{R}^2$, la section du câble. Les $N + 1$ conducteurs présents dans la section seront notés $(D_k)_{k=0,\dots,N}$, nous supposons que les conducteurs ne se touchent pas. Le conducteur 0 sera appelé référence de masse, dans le cas d'un domaine borné, le bord de ce conducteur sera le bord du domaine. La permittivité du milieu présent dans la section est notée ε . ε est un champ scalaire constant par morceaux. Nous supposerons que chaque conducteur est entouré par un milieu de conductivité constante. Enfin, nous notons Γ les discontinuités de la permittivité. Les discontinuités et les bords des conducteurs sont munis d'une normale unitaire n pointant vers l'intérieur des conducteurs (voir la figure FIGURE 10.2 page 173).

Nous expliciterons plus loin les formes prises par les conducteurs et les discontinuités.

Le potentiel électrique V est supposé \mathcal{C}^2 par morceaux.

Domaine borné homogène

Dans un premier temps, considérons un domaine Ω borné, comme représenté sur la figure FIGURE 10.2 page 173, et supposons la permittivité constante.

Théorème 5. *Soit*

$$G : \begin{array}{l} \mathbb{R}^2 \rightarrow \\ (x, y) \mapsto \frac{-1}{2\pi} \ln \left(\sqrt{x^2 + y^2} \right) \end{array} \quad (10.10)$$

G est solution fondamentale de l'équation de Laplace, i.e. au sens des distributions

$$-\Delta G = \delta \quad (10.11)$$

où δ est la mesure de Dirac.

Démonstration. En dehors de l'origine $\Delta G = 0$.

Par définition du laplacien au sens des distributions,

$$\forall \varphi \in \mathcal{D}(\mathbb{R}^2), \langle \Delta G, \varphi \rangle = \int_{\mathbb{R}^2} G \Delta \varphi. \quad (10.12)$$

Soit B_η la boule de centre 0 et de rayon $\eta > 0$. Comme G est dans L^1_{loc} , nous avons aussi

$$\forall \varphi \in \mathcal{D}(\mathbb{R}^2), \langle \Delta G, \varphi \rangle = \lim_{\eta \rightarrow 0^+} \int_{\mathbb{R}^2 \setminus B_\eta} G \Delta \varphi. \quad (10.13)$$

Nous notons alors n la normale unitaire sur ∂B_η pointant vers l'origine. Une double intégration par parties donne

$$\int_{\mathbb{R}^2 \setminus B_\eta} G \Delta \varphi = \int_{\partial B_\eta} \frac{\partial \varphi}{\partial n} G - \varphi \frac{\partial G}{\partial n} + \int_{\mathbb{R}^2 \setminus B_\eta} \varphi \Delta G. \quad (10.14)$$

En passant en coordonnées polaires, nous voyons alors que

$$\lim_{\eta \rightarrow 0^+} \int_{\partial B_\eta} \frac{\partial \varphi}{\partial n} G = 0, \quad (10.15)$$

puis que

$$\lim_{\eta \rightarrow 0^+} \int_{\partial B_\eta} \frac{\partial G}{\partial n} \varphi = -\varphi(0). \quad (10.16)$$

Donc

$$\begin{aligned} \langle -\Delta G, \varphi \rangle &= \varphi(0) \\ &= \langle \delta, \varphi \rangle. \end{aligned} \quad (10.17)$$

□

Nous pouvons alors exprimer le potentiel électrique sous la forme d'une intégrale, cette formulation est appelée potentiel de double couche.

Théorème 6. *Le potentiel électrique V vérifie*

$$\int_{\partial\Omega} (n \cdot \nabla V(y))G(x-y) - V(y)(n \cdot \nabla G(x-y)) = V(x) \begin{cases} 1 & , \text{ si } x \in \Omega \\ 1/2 & , \text{ si } x \in \partial\Omega \\ 0 & , \text{ sinon.} \end{cases} \quad (10.18)$$

Domaine borné inhomogène

Supposons maintenant que le domaine Ω est borné. La permittivité n'est plus supposée constante. Notons Ω_k les sous-domaines sur lesquels la permittivité est une constante valant ε_k . Sur chacun des sous-domaines, le potentiel vérifie la formulation intégrale (10.18).

En additionnant les contributions des différents sous-domaines, nous en déduisons une représentation dite de potentiel double couche. Les indices L et R correspondent aux valeurs prises de part et d'autre d'une discontinuité de permittivité.

Théorème 7. *Le potentiel électrique V vérifie*

$$\int_{\partial\Omega} (n \cdot \varepsilon \nabla V(y))G(x-y) - \int_{\Gamma} (\varepsilon_L - \varepsilon_R)V(y)(n \cdot \nabla G(x-y)) \\ = V(x) \begin{cases} \varepsilon_k & , \text{ si } x \in \Omega_i \\ (\varepsilon_L + \varepsilon_R)/2 & , \text{ si } x \in \Gamma \\ \varepsilon_L & , \text{ si } x \in \partial\Omega \\ 0 & , \text{ sinon.} \end{cases} \quad (10.19)$$

Démonstration. Le potentiel vérifie

$$\begin{aligned}
& \int_{\partial\Omega} (n \cdot \varepsilon \nabla V(y)) G(x-y) - \varepsilon V(y) (n \cdot \nabla G(x-y)) + \\
& \int_{\Gamma} (n \cdot (\varepsilon_L - \varepsilon_R) \nabla V(y)) G(x-y) - (\varepsilon_L - \varepsilon_R) V(y) (n \cdot \nabla G(x-y)) \\
& = V(x) \begin{cases} \varepsilon_k & , \text{ si } x \in \Omega_i \\ \varepsilon_L + \varepsilon_R / 2 & , \text{ si } x \in \Gamma \\ \varepsilon_L / 2 & , \text{ si } x \in \partial\Omega \\ 0 & , \text{ sinon.} \end{cases} \quad (10.20)
\end{aligned}$$

Tout d'abord, le terme $n \cdot (\varepsilon_L - \varepsilon_R) \nabla V$ est nul d'après la condition de saut vérifiée par le potentiel.

Ensuite, puisque le potentiel est constant sur les conducteurs, la quantité $\varepsilon_L V$ est constante sur le bord du domaine $\partial\Omega$. En appliquant la représentation du potentiel de double couche à un champ scalaire constant égal à $\partial\Omega$, nous obtenons

$$\int_{\partial\Omega} \varepsilon_L V (n \cdot \nabla G) = \varepsilon_L V \begin{cases} 1 & , \text{ si } x \in \Omega \\ 1/2 & , \text{ si } x \in \partial\Omega \\ 0 & , \text{ sinon.} \end{cases} \quad (10.21)$$

Ainsi, en injectant ce résultat dans la formule précédente, nous obtenons le résultat recherché. \square

Domaine non borné

Afin de traiter un domaine ouvert, nous postulons que la formule (10.19) reste valide. Cela équivaut à considérer que le potentiel à l'infini est nul. Même si ce résultat n'est pas démontré, cette formulation conduit à des résultats valides.

10.2.2 Discrétisation

Nous allons maintenant discrétiser la formulation obtenue dans les sections précédentes. Nous ne considérons ici que deux types de géométrie. Les contours des conducteurs et les discontinuités de permittivité sont soit des cercles soit des polygones.

Sur chaque contour présent dans la section étudiée, nous définissons un espace d'approximation contenant le potentiel électrique ainsi que sa dérivée normale.

Sur les contours des conducteurs ($\partial\Omega$), définissons la quantité $\sigma = \varepsilon_L n \cdot \nabla V$. Cette quantité correspond aux charges par unité de longueur présente sur le conducteur. De la même manière, nous définissons $\varphi = \frac{\varepsilon_L + \varepsilon_R}{2} V$ sur les discontinuités de permittivité Γ .

Avec cette notation, le problème s'écrit

$$\int_{\partial\Omega} \sigma(y)G(x-y) - \int_{\Gamma} \kappa\varphi(y)(\partial_n G(x-y)) = \begin{cases} \varepsilon_L V(x) & , \text{ si } x \in \partial\Omega \\ \varphi(x) & , \text{ si } x \in \Gamma \end{cases} \quad (10.22)$$

avec $\kappa = 2\frac{\varepsilon_L + \varepsilon_R}{\varepsilon_L - \varepsilon_R}$ (qui existe puisque défini sur les discontinuités de permittivité).

Soit un contour circulaire, jouant le rôle d'un conducteur, s'il constitue une limite du domaine de calcul, ou d'une discontinuité s'il est inclus dans le domaine de calcul. Soit $x_0 \in \mathbb{R}^2$ son centre et r son rayon. Un point sur ce cercle est alors repéré par l'angle θ formé entre l'axe $(0x)$ et le vecteur $\vec{x_0x}$

$$x = x_0 + r \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}. \quad (10.23)$$

Nous définissons alors sur ce cercle les $m_1 + m_2$ fonctions de base

$$\begin{cases} s_m : \theta \rightarrow \cos(m\theta) & , \text{ pour } 0 \leq m \leq m_1 \\ c_m : \theta \rightarrow \sin(m\theta) & , \text{ pour } 1 \leq m \leq m_2. \end{cases} \quad (10.24)$$

Soit un contour polygonal dont les sommets (S_0, \dots, S_M) orientés dans le sens direct sont donnés. Sur chaque segment $[S_k, S_{i+1}]$, nous définissons p_k sous-segments de mêmes longueurs

$$I_{i,m} = [S_k + \frac{m}{p_k}(S_{i+1} - S_k), S_k + \frac{m+1}{p_k}(S_{i+1} - S_k)] \quad (10.25)$$

pour $m = 0, \dots, p_k - 1$.

Les fonctions de base f_m sont alors les indicatrices de ces sous-segments.

Notons alors $(\psi_m^c)_{m=0, N_c}$ les N_c fonctions de base sur les conducteurs et $(\psi_m^d)_{m=0, N_d}$ les N_d fonctions de base sur les discontinuités de permittivité.

Nous définissons, de plus, des points contrôles sur $\partial\Omega$ et Γ . Sur les contours circulaires et polygonaux nous prenons un nombre de points de contrôles équirépartis égal au nombre de fonctions de base sur ce contour. Ainsi, cette discrétisation conduit à un système linéaire carré. Nous notons alors $(x_m^c)_{m=0, N_c}$ et $(x_m^d)_{m=0, N_d}$ les points de contrôle sur les conducteurs et les discontinuités.

Cette approche est un cas particulier de choix de fonctions tests. En effet, l'équation intégrale obtenue est multipliée par des fonctions tests et intégrée. Dans notre cas, les « fonctions » tests sont des masses de Dirac situées sur les points de contrôles.

Le système linéaire que nous devons résoudre peut alors s'écrire

$$\begin{pmatrix} M_{cc} & M_{cd} \\ M_{dc} & M_{dd} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} V_c \\ 0 \end{pmatrix}. \quad (10.26)$$

α , respectivement β , est le vecteur contenant les composantes de la charge σ , respectivement φ , dans la base d'approximation. Le second membre V_c contient les valeurs du potentiel aux points de contrôles sur les conducteurs.

Les quatre sous-matrices contiennent

$$\begin{aligned} M_{cc} &= \left(\int_{\partial\Omega} \psi_k^c(y) G(x_j^c - y) \right), \\ M_{cd} &= \left(- \int_{\Gamma} \kappa \psi_k^d(y) \partial_n G(x_j^c - y) \right), \\ M_{dd} &= \left(\int_{\partial\Omega} \psi_k^d(y) G(x_j^d - y) \right), \\ M_{dc} &= \left(\int_{\Gamma} \kappa \psi_k^d(y) G(x_j^d - y) - \psi_k^d(x_j^d) \right). \end{aligned} \quad (10.27)$$

En réalité, les intégrales précédentes ne sont faites que sur le contour portant la fonction de base intégrée.

En éliminant le vecteur β , nous obtenons une expression de α

$$\alpha = (M_{cc} - M_{cd}M_{dd}^{-1}M_{dc})^{-1} V. \quad (10.28)$$

En choisissant le second membre afin d'obtenir un potentiel valant un sur un contour et zéro sur les autres, nous pouvons calculer les charges associées. Par exemple, pour obtenir les charges q_j^k sur les conducteurs D_j engendrées par un potentiel nul sur tous les conducteurs excepté le $k^{\text{ième}}$, nous prenons un second membre V_k défini par

$$V_k = \left. \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \right\} \begin{array}{l} \text{valeurs correspondant aux points de} \\ \text{contrôles présents sur le conducteur } k. \end{array} \quad (10.29)$$

En appliquant la formule (10.4 page 173), nous calculons les charges q_j^k des conducteurs et construisons la matrice \mathcal{C} appelée matrice de capacitance généralisée

$$\mathcal{C} = \begin{pmatrix} q_0^0 & \dots & q_0^N \\ \vdots & & \vdots \\ q_N^0 & \dots & q_N^N \end{pmatrix}. \quad (10.30)$$

En suivant la procédure décrite dans [54], nous en déduisons les matrices de capacitance et inductance en ayant choisi un conducteur de référence.

Les différents contours présents dans une section peuvent avoir une longueur très différente. Chaque contour est muni d'un nombre de fonctions de base de manière à ce que le rapport entre sa longueur et ce nombre soit homogène sur toute la section.

10.3 Parallélisation

La résolution du problème discret décrit précédemment est effectuée en appliquant un algorithme parallèle utilisant la bibliothèque OpenCL. Le calcul des matrices de capacitance et inductance est alors séparé en plusieurs étapes. Dans un premier temps, les matrices M_{cc} , M_{cd} , M_{dc} et M_{dd} sont calculées par différents *kernels*. Dans un second temps, la matrice d'inductance est calculée en résolvant un système linéaire

$$M_{cc}\alpha = V. \quad (10.31)$$

où V joue le rôle de flux magnétique et α joue le rôle du courant et est construit comme en (10.29 page précédente).

Ensuite, si des discontinuités de diélectriques sont présentes, nous résolvons un second système linéaire (10.28 page précédente).

Les deux résolutions de systèmes linéaires font intervenir plusieurs seconds membres : un par conducteurs. Nous concaténons tous ces seconds membres en une matrice et résolvons en une seule fois les systèmes linéaires au moyen d'un algorithme de pivot de Gauss parallèle.

10.3.1 Calcul des matrices

Les différentes matrices M_{cc} , M_{cd} , M_{dc} et M_{dd} sont remplies de manière parallèle. Pour chaque matrice, un *work-item* calcule un terme associé à un indice i et un indice j de la matrice.

Les intégrales apparaissant dans les expressions des matrices ne sont pas approximées. Ces intégrales peuvent être calculées exactement.

En terme de temps de calcul, cette étape représente environ 5% du temps de calcul total. Néanmoins, il est intéressant de la réaliser sur le GPU afin d'éviter des transferts par le bus PCI. Ensuite, la majeure partie du temps est passée dans le traitement des systèmes linéaires.

10.3.2 Pivot de Gauss parallèle

Lors de la réalisation de cette étude (automne 2011), il n'existait pas de bibliothèque fiable et efficace d'algèbre linéaire pour OpenCL. Afin de ré-

soudre le système linéaire issu de la discrétisation de notre problème de départ, nous avons donc programmé une version parallèle de l'algorithme du pivot de Gauss pour OpenCL. Aujourd'hui, nous utiliserions sans doute une bibliothèque comme ViennaCL présentée dans [63].

Soit M une matrice de $\mathcal{GL}_n(\mathbb{R})$ et B une matrice de $\mathcal{M}_{nm}(\mathbb{R})$. Nous allons maintenant décrire les étapes permettant de calculer la matrice X de $\mathcal{M}_{nm}(\mathbb{R})$ telle que $MX = B$. Ces étapes suivent l'algorithme classique du pivot de Gauss. Pour commencer, les matrices sont stockées dans un unique tableau N de taille $n(n+m)$ de manière à ce que le terme M_{ij} sur la ligne i et la colonne j corresponde au terme d'indice $i(n+m) + j$ dans le tableau N . Pour les mêmes raisons qu'au chapitre 5 page 89, les indices des tableaux commencent à zéros. Nous avons choisi ce type de stockage (*column-first*) car les traitements demandant le plus d'accès mémoire requièrent la lecture de ligne ou partie de ligne. Ainsi ce rangement assure une lecture coalescente des données et limite donc les temps d'accès mémoire (voir la section 5.1.1).

Dans cette section, nous considérerons indifféremment le tableau N comme un tableau à $n(n+m)$ valeurs et comme une matrice à n lignes et m colonnes.

Nous appliquerons cet algorithme pour résoudre le système (10.28 page 181). Dans ce cas, la matrice M est égale à $M_{cc} - M_{cd}M_{dd}^{-1}M_{dc}$, le vecteur des inconnues X est égale à α et le second membre B contient les seconds membres décrits en (10.29 page 181).

Recherche du pivot maximal et élimination

Dans un premier temps, nous éliminons les valeurs sous-diagonales de M . Supposons que les valeurs sous-diagonales de M des i premières colonnes soient nulles et que ses i premiers termes diagonaux valent 1. Éliminons alors les termes sous-diagonaux de la $i + 1^{\text{ième}}$ colonne. Nous commençons par copier les valeurs sous-diagonales de la colonne $i + 1$ dans un tableau annexe P . Cette lecture est le seul accès mémoire non coalescent. Afin, de n'obtenir que des accès coalescents, il faudrait transposer la matrice N . Cependant, au vu de la taille de cette matrice comparée au nombre de lecture non coalescente, le gain obtenu ne serait pas significatif.

Afin de rechercher le pivot max, nous appliquons un algorithme de réduction parallèle (voir [5]) afin d'extraire la valeur maximale p_{max} du tableau P ainsi que son indice i_{max} dans le tableau. Pour cela nous définissons un entier p tel que $2^p \geq n$. Nous appliquons alors l'algorithme parallèle décrit en 10.1

avec pour opération de réduction l'application

$$x, y \mapsto \begin{cases} x & , \text{ si } |x| > |y| \\ y & , \text{ sinon.} \end{cases} \quad (10.32)$$

À chaque itération de cet algorithme, deux moitiés d'un sous-tableau de P sont comparées terme à terme. Le résultat de l'opération de réduction est alors conservé dans la première moitié du tableau. L'itération suivante traite alors un tableau de taille deux fois plus petite. Ainsi, après la dernière itération, le premier terme du tableau contient la valeur cherchée. Cet algorithme permet de réduire un tableau en n'effectuant que des lectures coalescentes et sans engendrer d'accès concurrents en mémoire.

Cet algorithme peut être modifié afin de maintenir un second tableau contenant l'indice dans le tableau initial de la valeur conservée. Pour cela, il suffit d'appliquer les mêmes opérations à ce second tableau.

```

__kernel void reduction(const int k,
                        const int size,
                        __global float* P){

    // Numero du workitem
    int i = get_global_id(0);

    float p1;
    float p2;

    // Boucle sur les sous-tableaux de P de taille
    // une puissance de 2.
    for(int level=k; level >= 1; level--){
        // Lecture de la premiere moitie du tableau.
        p1 = P[i];

        // 2 puissance level - 1
        int two_pow_l = 1 << (level - 1);

        // Lecture de la seconde valeur si
        // - le workitem est actif (i < two_pow_l)
        // - la valeur existe (i+two_pow_l < size)
        p2 = 0;
        if(i < two_pow_l && (i+two_pow_l) < size){

```

```

    p2 = P[i +two_pow_1];
}

barrier(CLK_GLOBAL_MEM_FENCE);

// Nous avons alors lue les deux moities du
// sous-tableau de P de taille 2^(level-1).

// Application de l'operation de reduction :
// operation.
if(i < two_pow_1){
    P[i] = operation(p1, p2);
}

barrier(CLK_GLOBAL_MEM_FENCE);
}
}

```

Algorithme 10.1 – Algorithme parallèle de réduction.

Nous échangeons alors les lignes $i + 1$ et i_{max} de N , afin de ramener le pivot le plus grand sur la diagonale. Cette opération s'effectue de manière parallèle en affectant à chaque *work-item* l'échange des valeurs correspondant à son numéro. Les lectures et écritures sont coalescentes. De plus, lors de cette opération, nous divisons la ligne i_{max} par la valeur du pivot p_{max} . Ceci permet d'économiser une lecture en mémoire.

Reste alors à éliminer les termes de la colonne $i + 1$ située sous la diagonale, *i.e.* à réaliser les opérations

$$N_{\alpha,\beta} = N_{\alpha,\beta} - N_{\alpha,i+1}N_{i+1,\beta}. \quad (10.33)$$

pour $i + 2 \leq \alpha \leq n$ et $i + 2 \leq \beta \leq n + m$

Pour cela nous affectons à cette opération $(n - i - 1)(n + m - i - 1)$ *work-items* indicés par deux entiers $i + 2 \leq \alpha \leq n$ et $i + 2 \leq \beta \leq n + m$. Un *work-item* lit alors les valeurs $N_{\alpha,\beta}$, $N_{\alpha,i+1}$ et $N_{i+1,\beta}$, puis effectue l'opération précédente. Les termes sous-diagonaux de N ne sont pas mis à jour. Leurs valeurs après cette opération sont connues à l'avance et ne seront donc plus utilisées. Nous prendrons simplement garde à ne plus les lire, mais à utiliser leurs valeurs connues *i.e.* 1 pour les termes diagonaux et 0 pour les termes sous-diagonaux.

Les étapes précédentes sont répétées jusqu'à élimination de tous les termes

sous-diagonaux. Comme dit précédemment, seuls les termes sur-diagonaux sont réellement écrits en mémoire, les autres termes ont des valeurs connues, mais non calculées.

Remontée

Cette étape consiste en l'élimination des termes sur-diagonaux de la matrice N . En partant de la dernière ligne et en remontant jusqu'à la première, nous annulons les termes placés au-dessus du terme diagonal dont la valeur est 1. Après la première itération, les termes de la colonne n de N sont tous nuls et les autres valeurs de la partie de N correspondant à M sont inchangés. En effet, les termes 1 à $n - 1$ de la dernière ligne de N sont tous nuls. De la même manière, après la $i^{\text{ième}}$ itération seules les valeurs de la colonne i seront modifiées ainsi que les valeurs de N correspondant à la matrice B .

Nous pouvons donc n'effectuer la remontée que sur la partie de N correspondant à la matrice B . Les autres valeurs n'ont pas d'importance dans le calcul et sont connues *a priori*, en fin de traitement la partie de N correspondant à M contiendrait l'identité.

Décrivons l'élimination des termes sur-diagonaux de la colonne i . Comme lors de la descente, nous affectons à cette opération $(n - i)m$ *work-items* indicés par deux entiers $1 \leq \alpha \leq n - i$ et $n \leq \beta \leq n + m$. Un *work-item* d'indice (α, β) effectuera alors l'opération suivante

$$N_{\alpha,\beta} = N_{\alpha,\beta} - N_{\alpha,i}N_{i,\beta}. \quad (10.34)$$

Les lectures et écritures mémoire sont dans ce cas coalescentes et ne génèrent pas de conflits d'écriture.

Une fois cette étape terminée, la partie de N correspondant à B contient la solution du système linéaire de départ.

10.4 Résultats

Dans cette section, nous donnons quelques résultats obtenus avec l'outil décrit dans ce chapitre. Dans un premier temps, nous étudions la convergence de la méthode dans un cas où l'on dispose d'une valeur exacte des matrices recherchées. Dans un second temps, nous nous intéresserons à un cas com-

plexe comportant un grand nombre de contours et comparons nos résultats et temps de calcul à ceux obtenus avec un outil de référence séquentiel appelé LaplaceCalcul et développé à l'ONERA.

10.4.1 Convergence

Considérons un cas simple pour lequel nous connaissons une solution exacte. Cette section est décrite sur la figure FIGURE 10.3 page suivante. Elle est constituée de deux conducteurs circulaires de rayons respectifs $r_1 = 4\text{mm}$ et $r_2 = 3\text{mm}$, dont les centres sont espacés d'une distance $d = 10\text{mm}$. Dans cette configuration la capacité est donnée par

$$C = \frac{2\pi\varepsilon}{\cosh^{-1}\left(\frac{d^2 - r_1^2 - r_2^2}{2r_1r_2}\right)} = 34.91\text{pF/m}. \quad (10.35)$$

Nous effectuons le calcul de la capacité de cette section en prenant différents nombres de fonctions de base et calculons l'erreur commise.

La figure FIGURE 10.4 page suivante représente le logarithme de l'erreur en fonction du nombre N de fonctions de base par conducteur. L'erreur e vaut alors

$$e(N) = O(\exp(N)^{-k}) \quad (10.36)$$

avec $k = 0.8$. On peut remarquer que l'erreur relative stagne à environ 10^{-9} .

10.4.2 Section complexe

Dans cette section, nous calculons la matrice capacité d'une section complexe composée de 80 conducteurs, 80 discontinuités de diélectrique et un plan de masse. Cela représente 5210 degrés de liberté. La figure FIGURE 10.5 page 189 représente les contours des conducteurs en bleu et des diélectriques en rouges. Ces contours sont placés au-dessus d'un plan de masse parfaitement conducteur.

Les matrices de capacitance et inductance sont alors calculées en utilisant l'outil présenté ici ainsi que par un autre outil dont les résultats serviront de référence.

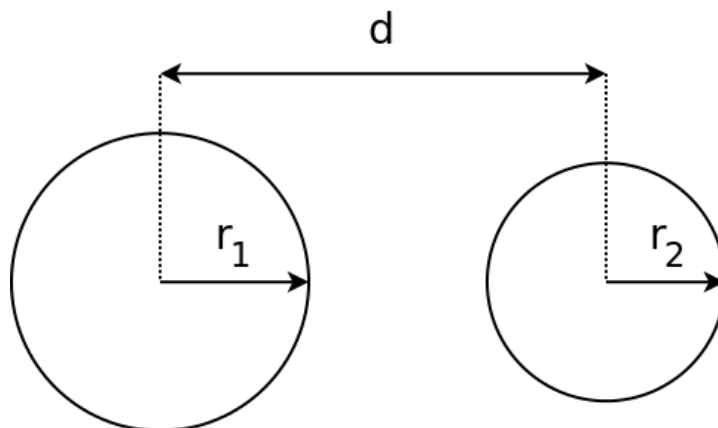


FIGURE 10.3 – Section de câble comportant deux conducteurs pour laquelle nous disposons d’une formule donnant la capacitance et l’inductance. Nous utilisons cette configuration dans l’étude de convergence de la méthode.

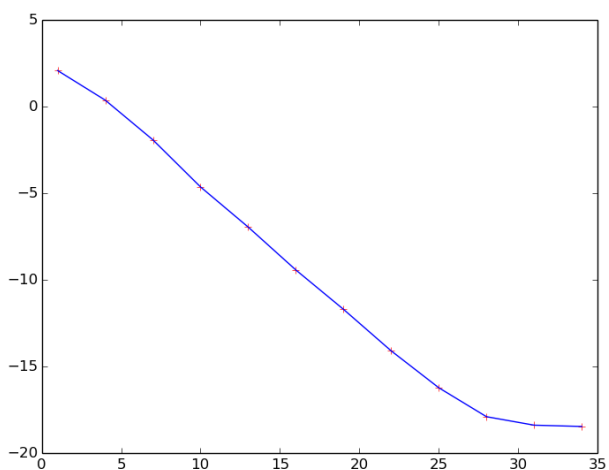


FIGURE 10.4 – Logarithme de l’erreur en fonction du nombre de fonctions de base lors du calcul de la capacitance de la section représentée sur la figure FIGURE 10.3.

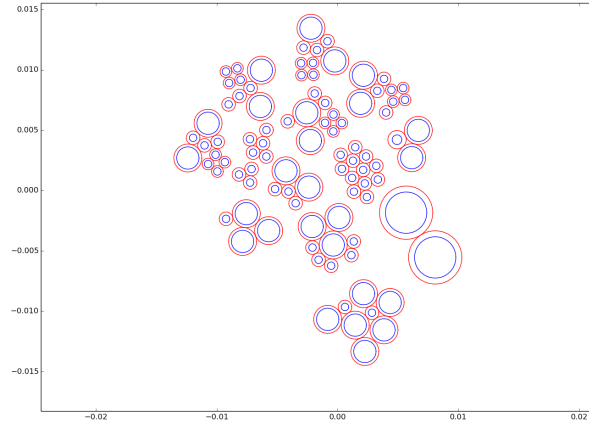


FIGURE 10.5 – Section complexe.

Nous effectuons les calculs en utilisant deux cartes graphiques différentes :

- Nvidia GeForce GTX 780 Ti,
- Nvidia GeForce GTX 460.

Les matrices obtenues diffèrent d'une valeur relative de 10^{-5} , ce qui valide notre calcul. Le temps de calcul obtenu avec l'algorithme séquentiel est de 19 minutes. L'implémentation OpenCL est entre 20 fois et 100 fois plus rapide (voir tableau 10.1). Les deux outils effectuent les calculs en double précision.

Carte graphique	Temps de calcul (s)	speedup
Nvidia GeForce GTX 780 Ti	10.6	114
Nvidia GeForce GTX 460	60.2	19

TABLE 10.1 – Temps de calcul de l'implémentation parallèle.

Conclusion

En conclusion, nous avons programmé un algorithme parallèle permettant de calculer les matrices de capacitance et inductance d'une section complexe de câble. Cet algorithme est au moins 20 fois plus rapide que l'algorithme séquentiel. Ce travail a été réalisé au début de la thèse pour une prise en main d'OpenCL. Nous avons présenté deux cas tests permettant de valider les

résultats fournis par cet outil. D'une part, nous avons vérifié la convergence de la méthode dans un cas simple où nous disposons d'une solution exacte. D'autre part, nous avons comparé les matrices calculées dans un cas complexe avec les résultats d'un code de référence implémentant la même méthode.

Conclusion générale

Cette thèse était consacrée à l'étude d'une méthode Galerkin Discontinue parallélisée sur cartes graphiques au moyen des bibliothèques OpenCL et MPI. Ces travaux s'inscrivent dans le cadre du projet GREAT financé par DGA/DS/MRIS dans le cadre d'un projet d'innovation duale RAPID. Ce projet a pour objectif d'industrialiser une méthode Galerkin Discontinue développée par l'ONERA et de la paralléliser.

Dans un premier temps, nous avons présenté les équations de Maxwell ainsi que quelques variantes de ces équations (équations de Maxwell avec correction de divergence, modèle PML). Nous avons montré que tous ces systèmes se mettent sous la forme d'un système hyperbolique linéaire symétrique, ou système de Friedrichs, et qu'il est donc important de disposer de méthodes numériques précises et optimisées pour les approcher.

Nous avons ensuite écrit une formulation faible associée à un système de Friedrichs quelconque, puis nous en avons déduit une formulation Galerkin Discontinue. Nous avons cherché à garder une certaine généralité dans notre approche afin de pouvoir utiliser l'outil que nous avons développé pour résoudre différents systèmes. La formulation GD que nous avons obtenue fait intervenir un flux numérique et un flux de bord. Dans cette étude, nous avons utilisé un flux décentré introduisant une dissipation qui améliore la stabilité du schéma. En étudiant l'énergie associée à la solution, nous avons déterminé une propriété devant être vérifiée par les flux de bord afin de garantir la stabilité du schéma.

Une fois la formulation GD établie, nous avons discrétisé la solution sur une base formée de produits tensoriels de polynômes de Lagrange. Les points d'interpolation sur lesquels sont construits ces polynômes sont les points de Gauss-Legendre. Ceux-ci nous ont également servi à définir des formules de quadrature permettant d'estimer les intégrales surfaciques et volumiques présentes dans la formulation GD. Ce choix de formules de quadrature et de base d'approximation permet également de simplifier les calculs des différents

termes entrant dans la formulation GD : la matrice de masse est diagonale, la matrice de rigidité est diagonale par blocs.

Comme nous l'avons mentionné, ces travaux ont pour but de produire un code industriel. Cela impose d'une part efficacité et fiabilité, mais également la possibilité de traiter une certaine variété de problèmes physiques. Nous avons donc mis en place différents modèles de matériaux et sources en nous basant sur les propriétés assurant l'existence et l'unicité de la solution et la stabilité du schéma. Ces différents modèles nous ont permis d'effectuer différentes simulations pour lesquelles nous disposons de comparaisons avec d'autres schémas.

La suite de ce manuscrit a été consacrée à l'implémentation parallèle de la méthode GD. Nous avons présenté une première approche dans laquelle le maillage est séparé en zone homogène traitée par un unique jeu de *kernels*. Nous avons programmé une version optimisée pour maillage hexaédrique de la méthode GD. Nous avons également parallélisé les calculs sur plusieurs cartes graphiques au moyen de la bibliothèque MPI. Dans cette version, nous communiquons entre sous-domaines, les données relatives à une couche de mailles volumiques. Nous avons ensuite procédé au test de cet outil. Ces tests ont permis de valider les résultats de la méthode et ont montré l'efficacité de la parallélisation sur carte graphique. Toutefois, la parallélisation MPI s'est révélée peu efficace, car le temps de transfert des données devient grand par rapport au temps de calcul.

Nous avons alors mis en place une seconde version de cette méthode GD. Dans celle-ci, le maillage est à nouveau séparé en zones homogènes, mais chacune d'entre elles est traitée par un jeu de *kernels* spécialisés. Cette structure de donnée pourra être enrichie de différents types mailles et d'interpolation. Actuellement, une interpolation basée sur les points de Gauss-Lobatto est développée. Nous avons également modifié la méthode de communication MPI. Dans ce cas, nous transférons uniquement des données surfaciques ce qui diminue fortement le volume de données à transférer. D'autre part, nous avons mis en place une stratégie d'asynchronisation des calculs et transfert de donnée nous permettant de continuer à effectuer des calculs durant les communications. Nous avons alors estimé l'efficacité de cette méthode en calculant la puissance de calcul fourni par l'outil en utilisant différents nombres de cartes graphiques. Nous avons alors constaté que nous obtenons une accélération proche de l'accélération théorique.

Dans deux courts chapitres, enfin, nous avons présentés des résultats que nous avons obtenus en parallèle du sujet principal de cette thèse :

- une nouvelle méthode pour éliminer la singularité du champ électromagnétique dans les couplages entre des modèles de fils et le modèle

- GD. Nous montr  sur des cas 1D tr s simples, que cette approche permet effectivement de supprimer le ph nom ne de Gibbs ;
- la description de la parall lisation OpenCL de l'algorithme de Gauss pour acc l rer (d'un facteur de l'ordre de 50) le calcul des matrices de capacitance et d'inductance dans les mod les de lignes de transmission.

Bibliographie

- [1] Hélène Barucq and Bernard Hanouzet. Etude asymptotique du système de maxwell avec la condition aux limites absorbante de silver-müller ii. Comptes rendus de l'Académie des sciences. Série 1, Mathématique, 316(10) :1019–1024, 1993.
- [2] Abderrahmane Bendali and Laurence Halpern. Conditions aux limites absorbantes pour le système de maxwell dans le vide en dimension 3. Comptes rendus de l'Académie des sciences. Série 1, Mathématique, 307(20) :1011–1013, 1988.
- [3] Jean-Pierre Berenger. A perfectly matched layer for the absorption of electromagnetic waves. Journal of computational physics, 114(2) :185–200, 1994.
- [4] JP Bérenger. Plaques minces aux différences finies. In 6ème Colloq. Int. Expos. Compat. Electromagn.(CEM), pages 298–303, 1992.
- [5] Guy E Blesloch. Prefix sums and their applications. Technical report, Carnegie Mellon University, 1990.
- [6] A. Bouquet. Caractérisation de structures rayonnantes par une méthode de type Galerkin Discontinu associée à une technique de domaine fictifs. PhD thesis, Université de Nice-Sophia Antipolis, 2007.
- [7] Françoise Bourdel, Pierre-Alain Mazet, Philippe Helluy, et al. Resolution of the non-stationary or harmonic maxwell equations by a discontinuous finite element method. application to an emi (electromagnetic impulse) case. In Proceedings of the 10th international conference on computing methods in applied sciences and engineering on Computing methods in applied sciences and engineering, pages 405–422, 1992.
- [8] Haïm Brezis. Analyse fonctionnelle, volume 5. Masson, 1983.
- [9] F. Brezzi, T. J. R. Hughes, L. D. Marini, and A. Masud. Mixed discontinuous galerkin methods for darcy flow. Journal of Scientific Computing, 22-23(1-3) :119–145, 2005.
- [10] Tristan Cabel, Joseph Charles, and Stephane Lanteri. Multi-GPU acceleration of a DGTD method for modeling human exposure to electromagnetic waves. Research Report RR-7592, Inria, April 2011.

- [11] Nicolas Castel, Gary Cohen, and Marc Duruffe. Application of discontinuous galerkin spectral method on hexahedral elements for aeroacoustic. Journal of Computational Acoustics, 17(02) :175–196, 2009.
- [12] Patrick Ciarlet, Jr. and Simon Labrunie. Numerical analysis of the generalized Maxwell equations (with an elliptic correction) for charged particle simulations. Math. Models Methods Appl. Sci., 19(11) :1959–1994, 2009.
- [13] Xavier Claeys. Asymptotics and numerical analysis for wave diffraction by thin wires. Theses, Université de Versailles-Saint Quentin en Yvelines, December 2008.
- [14] J.C. Clements, Clayton R. Paul, and A.T. Adams. Computation of the capacitance matrix for systems of dielectric-coated cylindrical conductors. Electromagnetic Compatibility, IEEE Transactions on, EMC-17(4) :238–248, Nov 1975.
- [15] Bernardo Cockburn and Chi-Wang Shu. Tvb runge-kutta local projection discontinuous galerkin finite element method for conservation laws. ii. general framework. Mathematics of Computation, 52(186) :411–435, 1989.
- [16] Gary Cohen, Xavier Ferrieres, and Sébastien Pernet. A spatial high-order hexahedral discontinuous galerkin method to solve maxwell’s equations in time domain. Journal of Computational Physics, 217(2) :340–363, 2006.
- [17] P.F. Combes. Micro-ondes : Tome 1, Lignes, guides et cavités. Micro-ondes. Dunod, 2006.
- [18] R. Courant and K.O. Friedrichs. Supersonic Flow and Shock Waves. Number vol. 21 in Applied mathematical sciences. Springer-Verlag, 1977.
- [19] Anaïs Crestetto. Optimisation de méthodes numériques pour la physique des plasmas. Application aux faisceaux de particules chargées. PhD thesis, Université Louis Pasteur - Strasbourg, 2012.
- [20] Anaïs Crestetto and Philippe Helluy. Resolution of the Vlasov-Maxwell system by PIC discontinuous Galerkin method on GPU with OpenCL. In CEMRACS’11 : Multiscale coupling of complex models in scientific computing, volume 38 of ESAIM Proc., pages 257–274. EDP Sci., Les Ulis, 2012.
- [21] Florian De Vuyst and Francesco Salvarani. Gpu-accelerated numerical simulations of the knudsen gas on time-dependent domains. Computer Physics Communications, 184(3) :532–536, 2013.
- [22] F. Edelvik. A new technique for accurate and stable modeling of arbitrarily oriented thin wires in the fdtd method. Electromagnetic Compatibility, IEEE Transactions on, 45(2) :416–423, May 2003.

- [23] Jiayuan Fang and Zhonghua Wu. Generalized perfectly matched layer for the absorption of propagating and evanescent waves in lossless and lossy media. Microwave Theory and Techniques, IEEE Transactions on, 44(12) :2216–2222, Dec 1996.
- [24] Loula Fezoui, Stéphane Lanteri, Stéphanie Lohrengel, and Serge Piperno. Convergence and stability of a discontinuous galerkin time-domain method for the 3d heterogeneous maxwell equations on unstructured meshes. ESAIM : Mathematical Modelling and Numerical Analysis, 39(06) :1149–1176, 2005.
- [25] Bruno Fornet, Vincent Mouysset, and Ángel Rodríguez-Arós. Mathematical study of a hyperbolic regularization to ensure Gauss’s law conservation in Maxwell-Vlasov applications. Math. Models Methods Appl. Sci., 22(4) :1150020, 28, 2012.
- [26] K. O. Friedrichs. Symmetric hyperbolic linear differential equations. Communications on Pure and Applied Mathematics, 7(2) :345–392, 1954.
- [27] S.D. Gedney. An anisotropic perfectly matched layer-absorbing medium for the truncation of fdtd lattices. Antennas and Propagation, IEEE Transactions on, 44(12) :1630–1639, Dec 1996.
- [28] Edwige Godlewski and Pierre-Arnaud Raviart. Numerical approximation of hyperbolic systems of conservation laws, volume 118 of Applied Mathematical Sciences. Springer-Verlag, New York, 1996.
- [29] Khronos group. Opencl. <https://www.khronos.org/opencl/>.
- [30] C. Guiffaut, P. Boudy, N. Rouvrais, B. Ben M’Hamed, B. Pecqueux, and A. Reineix. New thin coated wire formalism for fdtd method. In Electromagnetics in Advanced Applications (ICEAA), 2011 International Conference on, pages 386–389, Sept 2011.
- [31] Philippe Helluy. Numerical resolution of the harmonic Maxwell equations by a discontinuous Galerkin finite element method. Theses, Ecole nationale supérieure de l’aéronautique et de l’espace, January 1994.
- [32] Philippe Helluy, Jonathan Jung, et al. Two-fluid compressible simulations on gpu cluster. 2014.
- [33] J. S. Hesthaven and T. Warburton. High-order nodal discontinuous Galerkin methods for the Maxwell eigenvalue problem. Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 362(1816) :493–524, 2004.
- [34] Jan S. Hesthaven and Tim Warburton. Nodal discontinuous Galerkin methods, volume 54 of Texts in Applied Mathematics. Springer, New York, 2008. Algorithms, analysis, and applications.

- [35] R. Holland and Larry Simpson. Finite-difference analysis of emp coupling to thin struts and wires. Electromagnetic Compatibility, IEEE Transactions on, EMC-23(2) :88–97, May 1981.
- [36] I. Jeffrey. Finite-Volume Simulations of Maxwell’s Equations on Unstructured Grids. PhD thesis, University of Manitoba, Winnipeg, Manitoba, Canada, 2011.
- [37] A. Johnen, J.-F. Remacle, and C. Geuzaine. Geometrical validity of curvilinear finite elements. Journal of Computational Physics, 233(0) :359 – 372, 2013.
- [38] C. Johnson and J. Pitkäranta. An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. Math. Comp., 46(173) :1–26, 1986.
- [39] Sébastien Jund. Méthodes d’éléments finis d’ordre élevé pour la simulation numérique de la propagation d’ondes. Université Louis Pasteur. Institut de Recherche Mathématique Avancée (IRMA), Strasbourg, 2007. Thèse, Université Louis Pasteur, Strasbourg, 2007.
- [40] A. Klöckner, T. Warburton, J. Bridge, and J. S. Hesthaven. Nodal discontinuous Galerkin methods on graphics processors. J. Comput. Phys., 228(21) :7863–7882, 2009.
- [41] Serguei S. Komissarov. Multidimensional numerical scheme for resistive relativistic magnetohydrodynamics. Monthly Notices of the Royal Astronomical Society, 382(3) :995–1004, 2007.
- [42] David A. Kopriva. Metric identities and the discontinuous spectral element method on curvilinear meshes. J. Sci. Comput., 26(3) :301–327, 2006.
- [43] David A. Kopriva and Gregor Gassner. On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods. J. Sci. Comput., 44(2) :136–155, 2010.
- [44] David A. Kopriva, Stephen L. Woodruff, and M. Y. Hussaini. Computation of electromagnetic scattering with a non-conforming discontinuous spectral element method. International Journal for Numerical Methods in Engineering, 53(1) :105–122, 2002.
- [45] Lilia Krivodonova and Ruibin Qin. An analysis of the spectrum of the discontinuous galerkin method. Applied Numerical Mathematics, 64(0) :1 – 18, 2013.
- [46] T. Lähivaara and T. Huttunen. A non-uniform basis order for the discontinuous Galerkin method of the 3D dissipative wave equation with perfectly matched layer. J. Comput. Phys., 229(13) :5144–5160, 2010.
- [47] P. Lasaint and P.-A. Raviart. On a finite element method for solving the neutron transport equation. Math. Res. Center, Univ. of Wisconsin-Madison, Academic Press, New York, 1974.

- [48] Matti Lassas, Jukka Liukkonen, and Erkki Somersalo. Complex riemannian metric and absorbing boundary conditions. Journal de Mathématiques Pures et Appliquées, 80(7) :739 – 768, 2001.
- [49] Sophie Laurens. High accuracy approximation for diffraction problems. Theses, Université Paul Sabatier - Toulouse III, March 2010.
- [50] P. D. Lax and R. S. Phillips. Local boundary conditions for dissipative symmetric linear differential operators. Comm. Pure Appl. Math., 13 :427–455, 1960.
- [51] Pierre-Alain Mazet, Stéphan Paintandre, and Adib Rahmouni. Interprétation dispersive du milieu PML de Bérenger. C. R. Acad. Sci. Paris Sér. I Math., 327(1) :59–64, 1998.
- [52] Pierre-Alain Mazet, Lionel Ségui, and Bilkite Dah. Sur l’existence et l’unicité des solutions pour le système de Maxwell harmonique en présence de couches de Bérenger. C. R. Acad. Sci. Paris Sér. I Math., 333(6) :599–604, 2001.
- [53] C.-D. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, and U. Voß. Divergence correction techniques for Maxwell solvers based on a hyperbolic model. J. Comput. Phys., 161(2) :484–511, 2000.
- [54] Clayton R. Paul and A.E. Feather. Computation of the transmission line inductance and capacitance matrices from the generalized capacitance matrix. Electromagnetic Compatibility, IEEE Transactions on, EMC-18(4) :175–183, Nov 1976.
- [55] C.R. Paul. Analysis of Multiconductor Transmission Lines. Wiley-Interscience. Wiley, 2008.
- [56] L. Pebernet, X. Ferrières, S. Pernet, B.L. Michielsen, F. Rogier, and P. Degond. Discontinuous galerkin method applied to electromagnetic compatibility problems : introduction of thin wire and thin resistive material models. Science, Measurement Technology, IET, 2(6) :395–401, November 2008.
- [57] Vesselin M. Petkov and Luchezar N. Stoyanov. Geometry of reflecting rays and inverse spectral problems. Pure and Applied Mathematics (New York). John Wiley & Sons, Ltd., Chichester, 1992.
- [58] Chris J Railton and Dominique L Paul. Analysis of structures containing sharp oblique metal edges in fdtd using mamps. Antennas and Propagation, IEEE Transactions on, 58(9) :2954–2960, 2010.
- [59] Chris J Railton, Dominique L Paul, Ian J Craddock, and Geoffrey S Hilton. The treatment of geometrically small structures in fdtd by the modification of assigned material parameters. Antennas and Propagation, IEEE Transactions on, 53(12) :4129–4136, 2005.
- [60] Jeffrey Rauch. Symmetric positive systems with boundary characteristic of constant multiplicity. Trans. Amer. Math. Soc., 291(1) :167–187, 1985.

- [61] P-A Raviart and JM Thomas. Primal hybrid finite element methods for 2nd order elliptic equations. Mathematics of computation, 31(138) :391–413, 1977.
- [62] W. J. Rider and R. B. Lowrie. The use of classical lax–friedrichs riemann solvers with discontinuous galerkin methods. International Journal for Numerical Methods in Fluids, 40(3-4) :479–486, 2002.
- [63] K. Rupp, F. Rudolf, and J. Weinbub. ViennaCL - A High Level Linear Algebra Library for GPUs and Multi-Core CPUs. In Intl. Workshop on GPUs and Scientific Applications, pages 51–56, 2010.
- [64] JS Savage, WT Smith, and CR Paul. Moment method calculation of the per-unit-length parameters of cable bundles. In Electromagnetic Compatibility, 1994. Symposium Record. Compatibility in the Loop., IEEE International Symposium on, pages 441–446. IEEE, 1994.
- [65] Laurent Schwartz. Méthodes mathématiques pour les sciences physiques, volume 3. Hermann, 1961.
- [66] T.B.A. Senior. Impedance boundary conditions for imperfectly conducting surfaces. Applied Scientific Research, Section B, 8(1) :418–436, 1960.
- [67] Franck Sonnemann, Jean-Christophe Joly, and Bernard Pecqueux. Vulnerability of a missile facing hpm attack. In European Survivability Workshop, 2004.
- [68] Dennis M. Sullivan. An unsplit step 3-d pml for use with the fdtd method. Microwave and Guided Wave Letters, IEEE, 7(7) :184–186, Jul 1997.
- [69] Allen Taflov and Susan C Hagness. Computational electrodynamics. The finite-difference Time-Domain method. Third edition. Artech house, 2005.
- [70] Samuel Williams, Andrew Waterman, and David Patterson. Roofline : an insightful visual performance model for multicore architectures. Communications of the ACM, 52(4) :65–76, 2009.
- [71] J.-P. Wrenger. Numerical reflection from fdtd-pmls : a comparison of the split pml with the unsplit and cfs pmls. Antennas and Propagation, IEEE Transactions on, 50(3) :258–265, Mar 2002.
- [72] B. Zhao. The application of Discontinuous Galerkin element time-domain method in the design, simulation and analysis of modern radio frequency systems. PhD thesis, University of Kentucky, 2011.

Résolution des équations de Maxwell tridimensionnelles instationnaires sur architecture massivement multicœur

Résumé

Cette thèse s'inscrit dans un projet d'innovation duale RAPID financé par DGA/DS/MRIS et appelé GREAT faisant intervenir la société Axessim, l'ONERA, INRIA, l'IRMA et le CEA. Ce projet a pour but la mise en place d'une solution industrielle de simulation électromagnétique basée sur une méthode Galerkin Discontinue (GD) parallèle sur maillage hexaédrique. Dans un premier temps, nous établissons un schéma numérique adapté à un système de loi de conservation. Nous pouvons ainsi appliquer cette approche aux équations de Maxwell, mais également à tout système hyperbolique. Dans un second temps, nous mettons en place une parallélisation à deux niveaux de ce schéma. D'une part, les calculs sont parallélisés sur carte graphique au moyen de la bibliothèque OpenCL. D'autre part, plusieurs cartes graphiques peuvent être utilisées, chacune étant pilotée par un processus MPI. De plus, les communications MPI et les calculs OpenCL sont asynchronisés permettant d'obtenir une forte accélération.

Mots-clés : Galerkin Discontinue, équations de Maxwell, OpenCL, MPI, parallélisation.

Abstract

This thesis is part of a dual innovation project funded by RAPID DGA/DS/MRIS and called GREAT involving Axessim company, ONERA, INRIA, IRMA and the CEA. This project aims at the establishment of an industrial solution of electromagnetic simulation based on a method Discontinuous Galerkin (DG) on parallel hexahedral mesh. First, we establish a numerical scheme adapted to a conservation law system. We can apply this approach to the Maxwell equations but also to any hyperbolic system. In a second step, we set up a two-level parallelization of this scheme. On the one hand, the calculations are parallelized on graphics card using the OpenCL library. On the other hand, multiple graphics cards can be used, each driven by a MPI process. In addition, MPI communications and OpenCL computations are launched asynchronously in order to obtain a strong acceleration.

Keywords: Discontinuous Galerkin, Maxwell equations, OpenCL, MPI, parallelization.