AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

UFR S.T.M.I.A.
École Doctorale IAE + M
Université Henri Poincaré - Nancy I
D.F.D. Mathématiques

Traitement de couche limite et méthodes PIC : Algorithmes et Approche Objet

UNIVERSITÉ NANCY 1 BIBLIOTHÈQUE SCIENCES

THÈSE

présentée et soutenue publiquement le 5 Janvier 2000

pour l'obtention du

Doctorat de l'Université Henri Poincaré – Nancy I

(Spécialité Mathématiques appliquées)

par

Laurence Viry

Composition du jury

Président :	M. Francis Conrad	Professeur, Université Henri Poincaré, Nancy 1
Rapporteurs :	M. Jean-Pierre Croisille	Professeur, Université de Metz
	M. Pierre Fabrie	Professeur, Université de Bordeaux 1
Examinateurs :	M. Michel Pierre	Professeur, ENS-Cachan, Rennes
	M. Marc Garbey	Professeur, Université de Lyon 1
	M. Eric Sonnendrucker	Chargé de recherche CNRS, IECN, Nancy 1
	M. Olivier Coulaud	Chargé de recherche INRIA Lorraine

Institut Élie Cartan Nancy

Abstract

The work exposed in this document is made up of two relatively independant parts.

- The first part deals with mathematical and numerical simulations for a model of high frequency electromagnetic shaping of liquid metals. The asymptotic analysis gives prominence to the existence of singular perturbation problems with a weak layer closeness of the conductor surface.

  From our numerical model simplified formulation emerge two kinds of singular perturbation problems, classical elliptic problem and elliptic problem with electromagnetic transmission. We study in details a domain decomposition without recovering based on finite difference framework in one dimensional space to solve the transmission problem. We show that, for a good choice of the domain decomposition and boundary conditions at the artificial interface, we obtain a uniform approximation of the solution and the iterative procedure convergence rate which is superlinear. Then, we generalize these results into a two dimensional space for the radial case and we apply our results to enhanced finite element computations. We obtained rigorous convergence and accuracy estimates which seems to indicate that these schemes could be extended to more general geometries.

- The second part deals with a problem in computer science from

  - a computer science point of view: underlining opportunities of the object-oriented paradigm for advanced P.I.C (Particle-In-Cell in plasma physic) modelling, with these known advantages : increased flexibility, extensibility, robustness and so on. A two-dimensional relativistic electromagnetic PIC code implementation has been made.
  - a numerical point of view : a comparative study about electrical field correction in electromagnetic PIC plasma simulations is done. We analyze two techniques for reducing electric field error: the first one consists in using a charge conservating method to compute charge and current density onto the Maxwell discretization mesh, the second one is to use one divergence correction of electric fields based on the reformulated Maxwell system.

Keywords: Maxwell, boundary layers, singular perturbations, transition layers, asymptotic analysis, domain decomposition, object-oriented programming, PIC, plasma simulations, current weighting, Gauss'law, divergence corrections, rigorous charge conserving algorithm

Résumé

Le travail exposé dans ce document se compose de deux parties relativement indépendantes.

- La première relève de l'analyse et de la simulation numérique d'un problème physique lié au formage électromagnétique en haute fréquence. Le comportement asymptotique du modèle met en évidence l'existence d'une couche limite au voisinage de la surface du conducteur.

  Une écriture simplifiée d'un modèle numérique de notre problème a permis de dégager deux sous-problèmes à perturbation singulière, un problème elliptique classique et un problème elliptique avec transmission du phénomène électromagnétique. Nous étudions de façon complète en dimension un d'espace une méthode de décomposition de domaine sans recouvrement basée sur un schéma de résolution en différences finies et nous montrons, que pour un bon choix de la décomposition de domaine et de bonnes conditions limites à la frontière du métal, l'algorithme fournit une approximation uniforme de la solution et la vitesse de convergence du schéma itératif est superlinéaire. Puis, nous généralisons à la dimension 2, dans le cadre de géométries circulaires. Une implémentation des algorithmes étudiés, utilisant une approximation de type éléments finis nous a fourni de bons résultats, quant à la précision de la solution obtenue et la vitesse de convergence. Ceci nous laisse penser que ces algorithmes devraient s'adapter à des géométries plus complexes.

- La seconde partie est plus de nature "calcul scientifique" avec
  - des aspects informatiques : par la mise en évidence de l'apport de la programmation orientée objet (P.O.O) pour répondre à des critères de qualité d'un code (exactitude, robustesse, extensibilité, réutilisabilité...), et l'implémentation d'un code PIC (Particle-In-Cell) de la physique des plasmas.

  - et des aspects plus numériques : par une étude comparative des méthodes de correction du champ électrique dans un modèle PIC. Une première approche consiste à faire un calcul des densités de telle façon que l'équation de conservation de la charge soit vérifiée. La seconde approche consiste à effectuer une correction des champs électriques après la résolution des équations de Maxwell.

Mots-clés: Maxwell, couche limite, perturbation singulière, analyse asymptotique, décomposition de domaine, programmation orientée objet, P.I.C, simulation de plasmas, interpolation de la densité de courant, loi de Gauss, correction de la divergence, conservation de charge exacte

Remerciements

Je voudrais tout d'abord exprimer mes remerciements à :

Michel Pierre, pour la confiance qu'il m'a accordée en m'intégrant dans son équipe, pour ses cours de DEA qui m'ont redonné goût aux mathématiques, pour ses chaleureux encouragements et la relecture attentive de ce document,

Marc Garbey qui m'a beaucoup appris sur la façon d'aborder et de traiter un problème de calcul scientifique, pour son encadrement tonique, ses encouragements, son enthousiasme et sa bonne humeur,

Eric Sonnendrucker, pour son approche multidisciplinaire des problèmes traités qui en fait un interlocuteur de choix, pour les discussions fructueuses, sa patience face à certaines de mes indisponibilités et pour son calme et sa gentillesse,

Olivier Coulaud, pour les divers éclaircissements concernant le modèle physique sur lequel s'est appuyé la première partie de mon travail.

Je remercie également :

Monsieur Francis Conrad, Professeur à l' Université Henri Poincaré qui a accepté de présider ma soutenance,

Monsieur Jean-Pierre Croisille, Professeur à l'Université de Metz et Monsieur Pierre Fabrie, Professeur à l' Université de Bordeaux 1 qui ont accepté d'être rapporteurs de ma thèse.

Enfin, je remercie tous les amis et collègues, ils sauront se reconnaître, qui ont su m'encourager dans les moments difficiles où mener de front travail, recherche et famille me semblait trop épuisant.

Biensur, je ne peux oublier mes deux fils Antoine et Benjamin qui ont du bon gré mal gré vivre les périodes de travail intense qui ont permis de mener à bien ce projet.

À mon père

Quand bien nous pourrions être savants du savoir d'autrui, au moins sages ne pouvons-nous être que de notre propre sagesse.
Montaigne

Table des matières

Résumé i

Abstract ii

Avant-propos

Chapitre 1 Modélisation du problème physique 1

1.1 Introduction 1

1.2 Les équations 2

1.2.1 Équations du champ magnétique 2

1.2.2 Équations du mouvement 4

1.2.3 Les conditions limites 4

Conditions métal-air 4

1.2.4 Récapitulatif 5

1.2.5 Mise adimensionnelle 6

1.3 Approximation hautes fréquences 8

1.3.1 Introduction 8

1.3.2 Analyse asymptotique - Aspect magnétique 8

Cas fixe 8

Modèle variable 11

Conclusion 14

1.3.3 Echelle à pas multiples 15

Dans le cas des conducteurs solides 15

Dans le cas des conducteurs liquides 15

Chapitre 2 Vers l'écriture simplifiée de notre modèle 17

2.1 Introduction 17

2.2 Équations de Maxwell 18
2.2.1 Formulation potentielle des équations de Maxwell 19
2.2.2 Discrétisation en temps 20
2.2.3 Écriture simplifiée du modèle 21
2.2.4 Résolution 22
2.3 Equations de Navier-Stokes 22
2.3.1 Formulation vitesse - pression 22
2.3.2 Formulations vitesse - tourbillon 23
2.3.3 Formulation ω-ψ des équations de Navier-Stokes 25
2.3.4 Discrétisation en temps 26
2.3.5 Écriture simplifiée du modèle simplifié 27
2.4 Frontière libre 28
2.4.1 Détermination de la courbure 29
Calcul de ∇p 29
Détermination de la frontière 29
2.5 Formulation simplifiée du modèle complet 30

Chapitre 3 A domain decomposition adapted to our problem **31**
3.1 Introduction 31
3.2 Boundary layers in a One Dimensional Space 33
3.2.1 Homogeneous Domain Decomposition 33
Dirichlet-Neumann scheme 33
Neumann-Dirichlet scheme 40
3.2.2 Heterogeneous domain decomposition 43
Asymptotic analysis 43
Numerical procedure 44
3.3 Boundary layers in a Two Dimensional Space 50
3.4 Applications 58
3.4.1 Problème de transmission simplifié 58
3.4.2 Stationary case 60
Spatial discrétization 61
Résolution of transfer problem 62
Remarks 62
3.4.3 Non-stationary case 64

Chapitre 4 Modèles et langages de programmation Application au développement d'un code PIC **69**

4.1 Introduction . . . 69
4.2 Modèle et langage de programmation . . . 70
4.2.1 Introduction . . . 70
4.2.2 Programmation Orientée-Objet . . . 71
4.2.3 Pourquoi la P.O.O? . . . 72
4.2.4 Choix du langage . . . 74
4.3 Du modèle physique vers un modèle objet adapté . . . 78
4.3.1 Introduction . . . 78
4.3.2 Modèle mathématique . . . 80
Remarques . . . 82
4.3.3 Modèle objet adapté . . . 83
Schéma général de l'application . . . 83
Les principales classes mises en jeu dans le modèle commun . . . 84
Traitement des particules entrantes et sortantes . . . 87
Traitement d'une itération en temps . . . 88
Quelques remarques . . . 89
4.4 Implémentation d'un code PIC . . . 90
4.4.1 Introduction . . . 90
4.4.2 Algorithme . . . 90
Intégration en temps des champs et localisation de leurs composantes 90
Résolution des équations de Maxwell avec conditions limites de Silver Muller . . . 91
Interpolation . . . 93
Mouvement des particules . . . 94
Traitement des particules entrantes et sortantes . . . 96
4.4.3 Implémentation . . . 96
Remarques: . . . 98

Chapitre 5 Correction du champ électrique dans un code PIC **101**

5.1 Introduction . . . 101
5.2 Méthodes de correction . . . 103
5.2.1 Introduction . . . 103
5.2.2 Méthodes respectant l'équation de conservation de la charge . . . 105

Calcul de la densité de courant 106
5.2.3 Méthode de Boris 110
Algorithme 110
Discrétisation 111
5.2.4 Méthodes de Marder-Langdon 112
Discrétisation 114
Modification de Langdon de la méthode de Marder 115
Remarques 117
5.2.5 Méthode hyperbolique 118
Discrétisation 120
5.3 Implémentation 121
5.4 Conclusion 123

Bibliographie **131**

Avant-propos

Le travail exposé dans ce document se compose de deux parties relativement indépendantes. La première relève de l'analyse et de la simulation numérique d'un problème physique lié aux traitements des métaux liquides; l'autre concerne le traitement des codes et relève plus de calcul scientifique (avec des aspects informatiques et numériques). Bien qu'indépendantes, ces deux parties sont historiquement et scientifiquement liées : la deuxième est en particulier, issue de réflexions générées lors de la mise en oeuvre de l'implémentation sur machine des algorithmes numériques introduits dans la première partie. Elle est aussi motivée par le traitement numérique et informatique de simulations numériques en physique des plasmas.

Commençons par décrire le contexte de la première étude. Son domaine d'application est celui des procédés industriels pour le traitement des métaux liquides utilisant les phénomènes d'induction magnétique. L'induction se caractérise, et se distingue des autres techniques électrothermiques, par sa capacité à injecter sans contact de l'énergie thermique (*effet Joule*) ou de l'énergie mécanique (*brassage*) dans les matériaux conducteurs de l'électricité. L'investigation expérimentale de tels procédés est souvent complexe et difficile à mettre en oeuvre. Aussi la simulation numérique apparaît-elle comme un outil intéressant pour décrire ces procédés.

On s'intéresse plus particulièrement au cas du formage électromagnétique. Lorsqu'un courant alternatif passe dans des inducteurs dans la région d'un conducteur liquide, il crée un courant induit dans ce conducteur, il y a création d'un champ magnétique qui génère des *forces de Lorentz*. Ces forces créent un mouvement qui peut être turbulent.

En courant alternatif de basse fréquence, les courants ont tendance à se répartir uniformément dans la section des conducteurs. Par contre, lorsqu'on élève la fréquence en régime sinusoïdal, on constate que les courants se localisent au voisinage de la surface des conducteurs sur une épaisseur typique appelée *épaisseur de peau*, qui dépend de la pulsation ω des courants et de la conductivité du matériau ([42]) et dont une estimation est donnée par la formule

$$\text{épaisseur de peau} = \delta \sim \sqrt{\frac{2}{\mu_0 \sigma \omega}}.$$

Ce phénomène est appelé **effet de peau**. Notons que les courants se concentrent sur une épaisseur de peau d'autant plus faible que la pulsation ω est élevée.

De nombreuses études ont été faites concernant les applications directement liées aux effets mécaniques des forces de Lorentz ([57]).

- *Brassage électromagnétique*: Les forces électromagnétiques induites dans un matériau électroconducteur liquide engendrent des écoulements recirculants turbulents,
- *formage électromagnétique des métaux liquide*, cas de l'aluminium ([80], [10], [9]),
- *contrôle des écoulements par un champ magnétique continu* ([64],[35],[34],[59]),
- *lévitation électromagnétique*: les forces électromagnétiques agissent sur toute la périphérie du conducteur qui se trouve fortement comprimée sous l'effet de la pression magnétique. La résultante de ces forces de pression peut équilibrer le poids d'un volume liquide qui peut être ainsi maintenu en lévitation. ([49],[67],[44],[63],[60]),
- *élaboration des matériaux*: frittage des poudres métalliques, fabrication de fibre de verre pour l'isolation, fusion en auto-creuset inductif,
- etc ...

Par son influence sur l'épaisseur de peau δ, la fréquence du courant alternatif imposé dans les inducteurs joue un rôle essentiel dans la sélection des effets induits dans toutes ces applications.

Ce travail s'inscrit dans l'objectif génèral, d'une part de mieux comprendre et quantifier les phénomènes physiques pour affiner la modélisation des procédés de traitement électromagnétiques des métaux liquides, d'autre part d'utiliser ces informations pour mettre au point des méthodes numériques pour le traitement des couches limites magnétiques et hydrodynamiques apparaissant dans ces procédés.

Le problème auquel nous nous sommes intéressés, correspond à un système qui comprend $N-1$ inducteurs Ω_2,...Ω_N, une région de métal liquide Ω_1 et une de vide ou d'air notée Ω_0. Le mouvement du métal liquide provient de l'action combinée du champ magnétique créé par les inducteurs au travers de la force de Lorentz et de la gravitation. On ne tiendra pas compte des courants induits dans les inducteurs.

Pour les hautes fréquences, l'étude du modèle physique montre que l'on doit considérer au moins deux échelles de temps, un temps court qui est le temps de pulsation magnétique, égal à l'inverse de la fréquence du courant, et un temps long qui est le temps caractéristique de diffusion du champ magnétique à l'intérieur du conducteur. Ce dernier ne dépend que des caractéristiques du matériau. A l'échelle du temps magnétique, on observe l'aspect magnétique, à l'échelle du temps de diffusion, on observe l'évolution de la forme du conducteur.

Le comportement asymptotique du modèle a été étudié en détail dans ([18]) pour les conducteurs solides et dans ([?]) pour les conducteurs liquides. Nous rappellerons les résultats obtenus lorsque nous présenterons le modèle physique. Dans les deux cas, ils montrent que nous sommes en présence d'un problème de perturbation singulière et qu'il existe une couche limite dans Ω_1 au voisinage de $\partial\Omega_1$. Comme énoncé plus haut, l'épaisseur de cette couche limite est inversement proportionnelle à la racine carré de la fréquence du courant.

Ainsi, pour les hautes fréquences, l'existence de cette couche limite implique des difficultés importantes pour la simulation numérique et requiert l'utilisation de schémas numériques efficaces.

Pour la mise en place du modèle numérique, on supposera l'existence d'une décomposition de l'induction magnétique en une partie oscillatoire (associée au temps court) et une partie non oscillatoire (associée au temps long) sous la forme $\underline{B}(x,t) = B(x,t)e^{i\omega t}$, ce qui, reporté dans les équations de Maxwell, permet d'éliminer la partie oscillatoire. On peut ainsi se ramener à des temps plus long et observer l'évolution du conducteur liquide.

Une écriture simplifiée d'un modèle numérique de notre problème a permis de dégager deux sous-problèmes à perturbation singulière intéressants en soi. L'un est un problème elliptique classique pour l'équation du tourbillon et la résolution du champ magnétique dans le métal. L'autre est du même type avec la transmission du phénomène électromagnétique à la frontière du métal liquide. Nous décrivons dans le deuxième chapitre les hypothèses et les choix qui ont été faits pour aboutir à cette écriture simplifiée.

Dans le chapitre suivant, nous faisons une étude complète en dimension un d'espace d'une méthode de décomposition de domaine sans recouvrement, basée sur un schéma de résolution en différences finies, pour une approximation numérique efficace des solutions. Puisqu'elle est sans recouvrement, cette méthode peut être utilisée pour résoudre l'équation du tourbillon ou l'équation du champ magnétique sur le métal mais elle peut l'être également pour résoudre le problème de transmission du phénomène électromagnétique. Nous montrons, que pour un bon choix de la décomposition de domaine et de bonnes conditions limites à la frontière du métal, l'algorithme fournit une approximation uniforme de la solution et la vitesse de convergence du schéma itératif est superlinéaire. Grâce à un lemme de comparaison, nous généralisons facilement les résultats obtenus à la dimension 2. Pour des raisons simplificatrices, nous nous sommes placés dans le cadre de géométries circulaires.

Dans le cas des conducteurs, nous réalisons une implémentation de l'algorithme étudié en utilisant une approximation de type différences finies avec Matlab. Une implémentation de ce même algorithme, dans le même contexte utilisant la méthodes des éléments finis avec la bibliothèque *Modulef* ([8]), nous a fourni des résultats comparables, quant à la précision de la solution obtenue et la vitesse de convergence, que ceux obtenus avec une approximation de type différences finies. Ceci nous laisse penser que cet algorithme devrait s'adapter à des géométries plus complexes.

Dans le cas des conducteurs liquides, l'algorithme que nous utilisons pour résoudre l'équation du tourbillon est identique à l'algorithme précédemment appliqué au cas stationnaire. Pourtant la réutilisation du code de façon simple et fiable s'est avérée lourde, la moindre modification du code existant entrainant des "effets de bord" parfois difficiles à contrôler. C'est ici que se situe la transition avec la deuxième partie de ce travail. Ces difficultés, de nature plus informatique, nous ont conduit à poursuivre une réflexion, parallèllement amorcée dans le cadre d'un travail lié aux fonctions d'ingénieur de l'auteur,

et ayant pour objet le choix de modèles de programmation et de langages les plus adaptés au développement d'applications de calculs scientifiques. Actuellement les réponses à ces questions sont très variées et sujettes à controverse.

Dans le chapitre 4, nous donnons des critères définissant un **code général**, puis nous décrivons comment les concepts de la programmation orienté objet (P.O.O) apportent des solutions pour satisfaire à ces critères, et comment peut se faire le choix du ou des langages de programmation dans le domaine du calcul scientifique. Ce travail s'est appuyé sur l'implémentation séquentielle et parallèle d'un code PIC (Particle-in-Cell) en physique des plasmas mais les arguments exposés dans le premier paragraphe peuvent s'adapter à l'implémentation de toute autre application de calcul scientifique.

Après un rappel de l'algorithme qui définit un code PIC, nous définissons ensuite le modèle objet de base associé et nous décrivons l'implémentation d'un programme d'électromagnétisme de la physique des plasmas en 2D utilisant les méthodes développées dans ([11]). Nous verrons également comment la conception objet va nous permettre de localiser la partie parallèle de l'implémentation de notre code dans le traitement des frontières en nous évitant une modification complète de notre code séquentiel.

Dans un code particulaire comme dans beaucoup d'autres simulations numériques, des approximations doivent être faites pour modéliser les problèmes physiques dans un temps raisonnable. Ces approximations peuvent induire des résultats qui s'éloignent de la physique. Dans le dernier chapitre, nous étudions différentes méthodes permettant de corriger les erreurs provenant principalement du calcul des densités de charge et de courant sur les noeuds du maillage de résolution des équations de Maxwell. L'équation de conservation de la charge n'est pas satisfaite de façon exacte ce qui entraine, sur des temps longs, des écarts à la loi de Poisson arbitrairement grand. Après avoir déterminé une solution de référence de notre problème, nous observons l'évolution des solutions par rapport à cette référence pour chacune de ces méthodes lorsque l'erreur à l'équation de Poisson diminue. L'implémentation de ces méthodes de correction est rapide et fiable dans notre code objet, elle se ramène à l'ajout de classes dérivées et de méthodes et grâce à l'encapsulation de données, les modifications restent locales sans "effet de bord" dans le reste du code.

Chapitre 1

Modélisation du problème physique

1.1 Introduction

Dans ce chapitre nous nous proposons de décrire les hypothèses physiques et les équations conduisant à la modélisation de l'induction magnétique pour le traitement des métaux liquides. Plus particulièrement, on s'intéresse au formage électromagnétique ou de guidage de jet.

Nous considérons un système de $N-1$ inducteurs Ω_2,...Ω_N parcourus par un courant alternatif de haute fréquence autour d'une région de métal liquide ou solide Ω_1 et une de vide ou d'air notée Ω_0 (voir figure 1.1). Dans le cas du métal liquide, son mouvement pro-

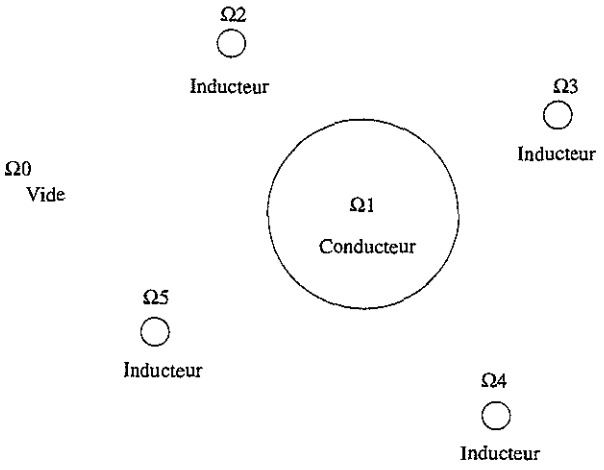


FIG. 1.1 – *Schéma conducteur-inducteurs (N=5)*

vient de l'action combinée du champ magnétique créé par le courant dans les inducteurs au travers de la force de Lorentz et de la gravitation.

Nous nous plaçons sous l'hypothèse suivante :

(H1) Le champ magnétique est régi par les équations de Maxwell, le mouvement du

métal par les équations de Navier-Stokes incompressibles. L'air est supposé à pression constante et non conducteur.

On note :

$$K = \bigcup_{i=2}^{N} \Omega_i,$$

avec

$$\Omega = \Omega_0 \cup K \qquad \text{et} \qquad \mathbb{R}^d = \Omega_1 \cup \Omega$$

où d est la dimension de l'espace (nous nous placerons en dimension 2 dans la partie simulation numérique du problème).

Dans ce chapitre, nous décrivons un modèle mathématique utilisé, adapté à une situation d'observation donnée, ainsi que les difficultés pour établir un modèle numérique efficace. Cette analyse met en évidence que le traitement du modèle complet (couplage Maxwell et Navier-Stokes) contient un certain nombre de difficultés :

- la mise adimensionnelle des équations de Maxwell et de Navier-Stokes nous indique qu'il faudra tenir compte en hautes fréquences d'un temps court qui représente la pulsation magnétique (inverse de la fréquence du courant dans les inducteurs, paramètre d'écran) et d'un temps long qui représente le temps de diffusion du champ d'induction dans le conducteur,
- le problème de transmission à la frontière du métal liquide dans le calcul du champ magnétique d'induction,
- les problèmes numériques liés à l'effet de peau dans le cas des hautes fréquences.

1.2 Les équations

Nous ne rappellerons pas ici comment sont obtenues les équations de Maxwell et de Navier-Stokes, pour plus de détails on se référera à ([42],[9]).

1.2.1 Équations du champ magnétique

Le champ magnétique est régi par les équations de Maxwell. Nous supposons que les matériaux de notre notre système satisfont aux trois hypothèses suivantes :

(H2) *Les matériaux, supposés homogènes et isotropes, ont un comportement linéaire qui se traduit par* :

$$B = \mu H \qquad \epsilon E = D$$

où ϵ est la perméabilité électrique, μ la perméabilité magnétique avec ϵ et μ constants dans les domaines Ω_i (i=0, ... N). On notera ϵ_0,μ_0 pour le domaine Ω_0 et ϵ_1,μ_1 pour le domaine Ω_1. Dans le vide, les constantes ϵ_0 et μ_0 ont pour valeurs dans le système mks-SI, $\mu_0 = 4\pi 10^{-7} Hm^{-1}$ et $\epsilon_0 = c^2/\mu_0 Fm^{-1}$ où c est la vitesse de la lumière.

(H3) Loi d'Ohm

$$J = \sigma E_t$$

où σ est la conductivité électrique, E_t le champ électrique total et J la densité de courant. Ils ont pour expressions :

$$\sigma(x) = \sigma_i \qquad x \in \Omega_i,\ i = 0, \dots N$$

$$E_t = \begin{cases} E & \text{dans les inducteurs et conducteurs solides,} \\ E + u \wedge B & \text{dans le métal liquide,} \\ E & \text{dans l'air ou les isolants magnétiques,} \end{cases}$$

où $u \wedge B$ est le champ électrique créé par le métal conducteur en mouvement dans le champ magnétique B.

(H4) *Les courants de déplacement sont négligeables devant les courants de conductions. Cette hypothèse est valable dès que* $\frac{\epsilon\omega}{\sigma} \ll 1$ *où* ω *est la fréquence du champ magnétique.*

On suppose les inducteurs multifilamentaires et soumis à une densité $\boldsymbol{J_d}$ donnée et **on ne tiendra pas compte des courants induits dans ces inducteurs.**

Définissons le prolongement J de la densité de courant dans l'air par :

$$J = \begin{cases} J_d & dans\ K \\ 0 & dans\ \Omega_0. \end{cases}$$

Le champ magnétique d'induction B est alors solution du système d'équations suivant :

$$\begin{cases} divB = 0 & \text{dans} \quad \Omega_1 \cup \Omega = \mathbb{R}^2 \\ \dfrac{\partial B}{\partial t} + \dfrac{1}{\mu_1 \sigma_1} Rot\, rot\, B = Rot(u \wedge B) & \text{dans} \quad \Omega_1 \\ rotB = \mu_0 J & \text{dans} \quad \Omega \end{cases} \tag{1.1}$$

où $\boldsymbol{u}$ est la vitesse du métal liquide.

Rappels :

- Soit $u = (u_1, u_2)$ une fonction régulière de $\mathbb{R}^2$ dans $\mathbb{R}^2$, on note $rot\, u$ la fonction à valeurs dans $\mathbb{R}$ définie par $rot\, u = \dfrac{\partial u_1}{\partial y} - \dfrac{\partial u_2}{\partial x}$.
- Soit ϕ une fonction régulière de $\mathbb{R}^2$ dans $\mathbb{R}$, on note $Rot\, \phi$ la fonction à valeurs dans $\mathbb{R}^2$ définie par $Rot\, \phi = (\dfrac{\partial \phi}{\partial y}, -\dfrac{\partial \phi}{\partial x})$.

1.2.2 Équations du mouvement

Sous l'hypothèse $(\boldsymbol{H_1})$, l'air est immobile et seul le métal liquide est en mouvement, celui-ci est régi par les équations de Navier-Stokes incompressibles suivantes :

$$\begin{cases} divu = 0 & \text{dans} \quad \Omega_1 \\ \rho(\dfrac{\partial u}{\partial t} + (u.\nabla)u) + \nabla p - \nu\Delta u = f & \text{dans} \quad \Omega_1 \end{cases} \tag{1.2}$$

où ρ est la masse volumique du métal, ν sa viscosité dynamique, u la vitesse, p la pression et f une densité volumique de force. On a :

$$f = f_g + f_L$$

où $\boldsymbol{f_g}$ est la partie gravitationnelle de f et $\boldsymbol{f_L}$ la partie magnétique appelée force de Lorentz et

$$f = \rho_g + \frac{1}{\mu_1} rotB \wedge B. \tag{1.3}$$

1.2.3 Les conditions limites

Conditions métal-air

- L'interface métal-air est représentée par l'équation :

$$F(x,t) = 0.$$

L'évolution de cette interface est régi par :

$$\frac{\partial F}{\partial t} + u.\nabla F = 0.$$

Cette équation est obtenue en imposant que *les particules sur la surface libre restent sur cette surface*, c'est une hypothèse courante dans les milieux continus.

Définissons la normale à F en (x,t) par :

$$n(x,t) = \frac{\nabla_x F}{\|\nabla_x F\|}.$$

On a :

$$u.n = -\frac{1}{< \nabla F.n >} \frac{\partial F}{\partial t} \quad = \quad v_n \tag{1.4}$$

où $\boldsymbol{v_n}$ est la vitesse normale de déplacement de l'interface.

- Nous avons la condition qui traduit *l'équilibre entre les contraintes et les forces de tensions superficielles*:

$$[\sigma.n]^{air}_{métal} = \upsilon C_{a,m}.n$$

où $\boldsymbol{\sigma}$ est le tenseur des contraintes, $\boldsymbol{\upsilon}$ la tension superficielle, $\boldsymbol{C_{a,m}}$ la courbure et $\mathbf{n}$ la normale extérieure à Ω_1.

Ce qui nous donne:

$$\begin{pmatrix} [\sigma.n].\tau \\ [\sigma.n]n \end{pmatrix} = \begin{pmatrix} 0 \\ \upsilon.C_{a,m} \end{pmatrix},$$

avec

$$\sigma = \rho I + Rem(\nabla u + \nabla u^T)$$

où Rem est le Reynolds magnétique.

- Pour le champ magnétique d'induction, nous avons:

$$\begin{cases} [B.n]^{air}_{métal} = 0 \text{ sur } \partial\Omega_1 \\ [H \wedge n]^{air}_{métal} = 0 \text{ sur } \partial\Omega_1. \end{cases}$$

- On a une décroissance à l'infini:

$$\lim_{\|x\| \to +\infty} \|B(x,t)\| = 0.$$

1.2.4 Récapitulatif

Résumons les équations avec les hypothèses (H1),(H2),(H3) et (H4) et la condition

$$\frac{\epsilon\omega}{\sigma} \ll 1.$$

Les équations dans Ω (extérieur du métal) sont:

$$\begin{cases} divB = 0 \\ rot(H) = J. \end{cases}$$

Dans Ω_1 (métal) on a:

$$\begin{cases} \dfrac{\partial B}{\partial t} + \dfrac{1}{\mu_1\sigma_1} RotrotB = rot(u \wedge B) \\ divB = 0 \\ divu = 0 \\ \rho(\dfrac{\partial u}{\partial t} + (u.\nabla)u) + \nabla p - \nu\Delta u = f, \end{cases}$$

avec

$$f = \rho_g + \frac{1}{\mu_1} rot(B) \wedge B.$$

Les conditions limites sur $\partial\Omega_1$:

$$\begin{cases} [B.n]^{air}_{métal} &= 0 \\ [H \wedge n]^{air}_{métal} &= 0 \\ u.n &= v_n \end{cases}$$

où v_n est la normale de la vitesse particulaire sur la frontière.

$$\begin{cases} (\nabla u + \nabla u^T).n.\tau &= 0 \\ p + Rem(\nabla u + \nabla u^T).n.n &= \upsilon C_{a,m}. \end{cases}$$

La condition limite à l'infini :

$$\lim_{\|x\| \to +\infty} \|B(x,t)\| = 0.$$

1.2.5 Mise adimensionnelle

Soient :

ω_0 : le maximum des fréquences de la densité de courant imposée dans les inducteurs,
J_0 : le maximum de la densité de courant imposée dans les inducteurs,
L_0 : le rayon de la sphère de volume V_0, volume initial du métal liquide,
P_0 : la pression,
U_0 : la vitesse.

On normalisera les coordonnées d'espace par L_0, le temps par $\frac{1}{\omega_0}$, le champ magnétique d'induction par $B_0 = \mu_0 J_0 L_0$, le champ magnétique par $H_0 = J_0 L_0$, le champ électrique par $E_0 = \omega_0 L_0 B_0$, la courbure par L_0^{-1} et la vitesse par U_0.

On verra plus loin, de façon plus précise, ce qui conditionne le choix du paramètre de normalisation en temps de nos équations.

On introduit :

- La fonction χ représentant les perméabilités relatives, définie par :

$$\chi(x) = \begin{cases} \frac{\mu_1}{\mu_o} & \text{sur } \Omega_1 \\ 1 & \text{sur } \Omega_0, \end{cases}$$

- le *nombre de Reynolds magnétique* **Rem** et le paramètre d'écran α définis par :

$$Rem = \mu_1 \sigma_1 L_0 U_0 \quad \text{et} \quad \epsilon^2 = \alpha = (\mu_1 \sigma_1 L_0^2 \omega_0)^{-1},$$

- $H_a = B_0 L_0 (\frac{\sigma_1}{\nu})^{-1}$: le nombre d'Hartman,

- $P_m = \frac{B_0^2}{\mu_o}$: la pression magnétique,

- $L_e = \frac{\sqrt{g L_0 \rho \mu_0}}{B_0}$: le nombre de lévitation,

- $E_m = \dfrac{P_m}{\rho U_0^2}$: le nombre d'Euler magnétique,
- $R_e = \dfrac{\chi H_a^2}{R_{em}}$: le nombre de Reynolds.

Avec les hypothèses (H1),(H2),(H3) et (H4) et la condition

$$\frac{\epsilon_1 \omega}{\sigma} \ll 1,$$

les équations de notre problème deviennent :

- Dans $\mathbb{R}^2$:

$$B = \chi H.$$

- Les équations dans Ω (extérieur du métal) sont :

$$\begin{cases} divB = 0 \\ rot(H) = J. \end{cases} \tag{1.5}$$

- L'équation du champ magnétique dans le métal devient :

$$\frac{\partial B}{\partial t} + \epsilon^2 RotrotB = \epsilon^2 rot(u \wedge B). \tag{1.6}$$

- La force de Lorentz :

$$\begin{aligned} f &= \mu_0 J_0^2 L_0 \chi^{-1} rot(B) \wedge B \\ &= \mu_0 J_0^2 L_0 f_L \\ &= \frac{B_0^2}{\mu_0 L_0} f_L \end{aligned} \tag{1.7}$$

où $\quad f_L = \chi^{-1} rot(B) \wedge B.$

En hautes fréquences, on prendra $f_L = \chi^{-1} rot(B) \wedge B$.

- Les équations de Navier-Stokes :

avec l'équation de la continuité :

$$divu = 0 \quad \text{dans} \quad \Omega_1, \tag{1.8}$$

et équation du mouvement :

$$\frac{1}{E_m}[\frac{1}{\epsilon^2 Rem}\frac{\partial u}{\partial t} + (u.\nabla)u] + \frac{P_0}{P_m}\nabla p - R_e^{-1}\Delta u = -L_e^2 e_z + f_L. \tag{1.9}$$

- Les conditions limites air-métal $\partial\Omega_1$:

$$\begin{cases} [B.n]^{air}_{métal} &= 0 \quad \text{sur} \quad \partial\Omega_1 \\ [H \wedge n]^{air}_{métal} &= 0 \quad \text{sur} \quad \partial\Omega_1 \\ u.n &= v_n \quad \text{sur} \quad \partial\Omega_1. \end{cases} \tag{1.10}$$

La condition d'équilibre des contraintes devient :

$$\overline{\sigma}_{ij} = \frac{B_0^2}{\mu_0}\sigma_{ij}, \text{ le tenseur des contraintes,}$$

avec

$$\sigma_{ij} = -\frac{P_0}{P_m}p\delta_{ij} + Re^{-1}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) + \chi^{-1}B_iB_j - \frac{1}{2\chi}(B.B)\delta_{ij}.$$

$$\begin{cases} [\sigma.n]^{air}_{métal} = \dfrac{\mu_0\gamma}{L_0B_0^2}C_{a,m}.n & \text{sur} \quad \partial\Omega_1 \\ (\nabla u + \nabla u^T).n.\tau = 0 & \text{sur} \quad \partial\Omega_1. \end{cases} \tag{1.11}$$

- La condition de décroissance à l'infini

$$\lim_{\|x\|\to+\infty} \|B(x,t)\| = 0. \tag{1.12}$$

1.3 Approximation hautes fréquences

1.3.1 Introduction

La mise adimensionnelle de notre problème a permis de mettre en évidence plusieurs échelles de temps. Le temps t_m de pulsation magnétique qui est égal à l'inverse de la fréquence du courant imposé dans les conducteurs (dans les applications considérées, il est de l'ordre de 10^{-3} à 10^{-4}s), le temps t_f de diffusion dans les conducteurs ($t_f = \frac{1}{\sigma_1\mu_1L_0^2}$, de l'ordre de 0.2s pour l'aluminium) et enfin le temps caractéristique t_v du mouvement du métal liquide ($t_m = \frac{U_0}{L_0}$, de l'ordre de la seconde pour l'aluminium).
Le paramètre ϵ^2 est le rapport du temps de pulsation magnétique et du temps caractéristique de diffusion dans le métal liquide et R_{em} (Reynolds magnétique) est le rapport du temps de diffusion et du temps caractéristique du mouvement du liquide. On peut donc se ramener à deux temps caractéristiques, un temps court t_m et un temps long t_f. A l'échelle du temps magnétique, on observe l'aspect magnétique, à l'échelle du temps de diffusion, on observe l'évolution de la forme du conducteur liquide.

1.3.2 Analyse asymptotique - Aspect magnétique

On montre par un approche basée sur une étude asymptotique classique ([16]) qu'au premier ordre, rien ne bouge dans le conducteur.

Cas fixe

On suppose que les conducteurs sont solides et fixes. Ce cas de figure correspond aux écrans magnétiques, les équations sont celles du métal avec $\mathbf{u}=\mathbf{0}$. On n'a pas de frontière libre.

On se ramène aux hypothèses et aux notations du premier paragraphe et on note $(P\epsilon)$ le système défini par :

$$(P_\epsilon)\begin{cases} \dfrac{\partial B}{\partial t} + \epsilon^2 RotrotB = 0 & \text{dans} \quad \Omega_1 \\ divB = 0 & \text{dans} \quad \mathbb{R}^d \\ B = \chi H & \text{dans} \quad \mathbb{R}^d \\ rotH = J & \text{dans} \quad \Omega \\ [B.n] = 0 & \text{sur} \quad \partial\Omega_1 \\ [H \wedge n] = 0 & \text{sur } \partial\Omega_1 \\ \lim\limits_{\|x\|\to+\infty} \|B(x)\| = 0 & \\ B(x,0) = \phi(x) & \text{donné dans } \Omega_1. \end{cases} \tag{1.13}$$

On cherche la solution B_ϵ sous la forme d'un développement en série de ϵ

$$B_\epsilon(x,t) = B_0(x,t) + \epsilon B_1(x,t) + o(\epsilon),$$

avec :

$$B_0(x,t) = \begin{cases} B_0^{ext}(x,t) & \text{si} \quad x \in \Omega \\ B_0^{int}(x,t) & \text{si} \quad x \in \Omega_1. \end{cases}$$

En reportant B_ϵ dans le système (P_ϵ), nous obtenons à l'ordre 0, le système suivant :

- $$\begin{cases} \dfrac{\partial B_0}{\partial t} = 0 & \text{dans} \quad \Omega_1 \\ divB_0 = 0 & \text{dans} \quad \mathbb{R}^d \\ B_0 = \chi H_0 & \text{dans} \quad \mathbb{R}^d \\ rotH_0^{ext} = J & \text{dans} \quad \Omega, \end{cases} \tag{1.14}$$

- des conditions limites qui doivent être vérifiées à tous les ordres

$$\begin{cases} [B_0.n] = 0 & \text{sur} \quad \partial\Omega_1 \\ [H_0 \wedge n] = 0 & \text{sur } \partial\Omega_1 \\ \lim\limits_{\|x\|\to+\infty} \|B_0(x)\| = 0, & \end{cases} \tag{1.15}$$

- les conditions initiales :

$$B_\epsilon(x,0) = B_0(x,0) + \epsilon B_1(x,0) = \phi(x) \quad \text{dans } \Omega_1, \tag{1.16}$$

ϕ est indépendant de ϵ.

Finalement à l'aide de (1.14) et (1.16), le champ magnétique d'induction vérifie :

$$B_0^{int} = \phi(x), \tag{1.17}$$

et B_0^{ext} satisfait le problème extérieur suivant :

$$(P_{ext}) \begin{cases} divB^{ext} = 0 & \text{dans } \Omega \\ rotH_0^{ext} = J & \text{dans } \Omega \\ B_0^{ext} = \chi H_0^{ext} & \text{dans } \Omega \\ B_0^{ext} = \phi(x) & \text{donné sur } \partial\Omega_1 \\ \lim\limits_{\|x\| \to +\infty} \|B_0^{ext}(x)\| = 0. \end{cases} \tag{1.18}$$

Remarques :

- le champ magnétique d'induction est indépendant du temps à l'ordre 0.
- La condition de Dirichlet sur $\partial\Omega_1$ est trop forte et rend le système (P_{ext}) surdéterminé. En effet, en affaiblissant la condition limite par $B_0^{ext}.n = h(x)$, on peut montrer que le problème (P_{ext}) admet une solution unique ([16]). Par conséquent, nous avons **un problème de perturbation singulière**, il existe une couche limite dans Ω_1 au voisinage de $\partial\Omega_1$.

Pour trouver un développement uniforme de B_ϵ, on va utiliser **la technique des développements raccordés**([16]). Pour cela, il nous faut construire B^{cl} dans la couche limite et (B^{ext},B^{int}) à l'extérieur de celle ci, ce qui nous donne à l'ordre 0 :

$$B_\epsilon(x,t) = \begin{pmatrix} B_0^{ext} \\ B_0^{int} \end{pmatrix} + \begin{pmatrix} 0 \\ B_0^{cl}(\gamma_1,\gamma_2,n,\epsilon,t) \end{pmatrix} + \epsilon B_1(x,t)$$

où (γ_1,γ_2,n) représente les coordonnées dans une base locale à la surface obtenue à partir d'une paramétrisation normale de $\partial\Omega$.

Construction dans la couche limite : Nous nous plaçons dans une approche bidimensionnelle et nous supposons que $\partial\Omega_1$ admet une paramétrisation normale. On note $\gamma(s)$ le vecteur unitaire tangent à $\partial\Omega_1$ en un point P(s) de la surface et par n(s) le vecteur unitaire normal extérieur à $\partial\Omega_1$ en P(s). On suppose qu'un point M de Ω_1 proche de la frontière va s'écrire de façon unique sous la forme :

$$M(x,y) = M(s,r) = P(s) - rn(s).$$

On pose

$$h_1 = 1 + \rho(s)r.$$

On écrit le système (P_ϵ) en coordonnées locales et le champ d'induction magnétique est décomposé en deux composantes, une tangentielle, B_s, et l'autre normale, B_n. Nous avons alors, $B_\epsilon = (B_s,B_n)$ et les résultats suivants([16]).

Proposition 1 *Le champ d'induction magnétique couche limite, $B_0^{cl} = (B_0^s,B_0^n)$ est solu-*

tion de

$$\begin{cases} B_n^0 = 0 \\ \dfrac{\partial B_s^0}{\partial t} - \dfrac{\partial^2 B_s^0}{\partial r^2} = 0 \\ B_s^0(s,0,t) = h(s,t) \\ B_s^0(s,r,0) \quad \textit{donnée} \\ \lim\limits_{r\to+\infty} B_s^0(s,r,t) = 0. \end{cases} \tag{1.19}$$

Proposition 2 *L'approximation uniforme de B_ϵ est donnée par*

$$B_\epsilon = B_0 + B_0^{cl} + o(1)$$

où $B_0 = (B_0^{ext}, B_0^{int})$ est solution de

$$\begin{cases} B_0 & = \chi H_0 & \textit{dans} & \mathbb{R}^d \\ div B_0^{ext} & = 0 & \textit{dans} & \Omega \\ rot H_0^{ext} & = J & \textit{dans} & \Omega \\ \lim\limits_{\|x\|\to+\infty} \|B_0^{ext}(x)\| & = 0 & & \\ B_0^{ext}(x) & = \phi(x).n, & & \end{cases} \tag{1.20}$$

et

$$B_0^{int}(x,t) = \phi(x).$$

L'approximation de couche limite $B_0^{cl} = (B_s^0, B_n^0)$ est donnée par

$$\begin{cases} B_n^0 = 0 \\ \dfrac{\partial B_s^0}{\partial t} - \dfrac{\partial^2 B_s^0}{\partial r^2} = 0 \\ B_s^0(s,0,t) = h(s,t) \\ B_s^0(s,r,0) \quad \textit{donnée} \\ \lim\limits_{r\to+\infty} B_s^0(s,r,t) = 0. \end{cases} \tag{1.21}$$

Remarque : Tout se passe à l'intérieur de la couche limite.

Modèle variable

Nous revenons au problème initial, on a un problème à frontière libre. Nous nous plaçons sous les hypothèses décrites dans le paragraphe 2 de ce chapitre. Notre problème possède deux temps caractéristiques, un temps court $\boldsymbol{t_m}$ et un temps long $\boldsymbol{t_f}$. Dans ce paragraphe, nous observons l'aspect magnétique, les équations sont adimensionalisées par $\frac{1}{\omega_0}$(temps court).

On notera $(P\epsilon)$ le système défini par (1.5), (1.6),(1.8),(1.9),(1.10),(1.11),(1.12).

Le problème est un problème à frontière libre, on note Ω_1^i le domaine Ω_1^ϵ à l'ordre i. Nous supposons que la frontière peut s'écrire sous la forme

$$F_\epsilon(x,t) = F_0(x,t) + \epsilon F_1(x,t). \tag{1.22}$$

On cherche B_ϵ,u_ϵ,p_ϵ sous la forme

$$\begin{array}{l} B_\epsilon(x,t) = B_0(x,t) + \epsilon B_1(x,t) \\ u_\epsilon(x,t) = u_0(x,t) + \epsilon u_1(x,t) \\ p_\epsilon(x,t) = p_0(x,t) + \epsilon p_1(x,t). \end{array} \tag{1.23}$$

On cherche B_i, B à l'ordre i, sous la forme

$$B_i(x,t) = \begin{cases} B_i^{ext}(x,t) & \text{si} \quad x \in \Omega^i(t) = \mathbb{R}^d - \Omega_1^i(t) \\ B_i^{int}(x,t) & \text{si} \quad x \in \Omega_1^i(t). \end{cases}$$

De la même façon, le tenseur des contraintes se décompose en

$$\sigma_\epsilon = \sigma_0 + \epsilon\sigma_1.$$

En reportant B_ϵ,u_ϵ,p_ϵ,F_ϵ dans (P_ϵ), nous obtenons l'approximation d'ordre 0 ([16]).

En première approximation à l'ordre 0 :

- la frontière libre est stationnaire, notre système n'est plus un problème à frontière libre et nous sommes ramenés à l'étude précédente en ce qui concerne le champ magnétique.
- Dans Ω_1^0 connu, B_0^{int} et la vitesse du métal sont stationnaires.
- Le champ B_0^{ext} est solution de

$$(P_{ext}) \begin{cases} div B_0^{ext} = 0 & \text{dans} \quad \Omega^0 \\ Rot H_0^{ext} = J & \text{dans} \quad \Omega^0 \\ B_0^{ext} = \chi H_0^{ext} & \text{dans} \quad \Omega^0 \\ B_0^{ext}.n = B_0^{int}.n = \phi_b^{int}.n & \text{donné sur} \quad \partial\Omega_1^0 \\ H_0^{ext} \wedge n = H_0^{int} \wedge n = \frac{1}{\chi}\phi_b^{int} \wedge n & \text{donné sur} \quad \partial\Omega_1^0 \\ \lim\limits_{\|x\| \to +\infty} \|B_0^{ext}(x)\| = 0 \end{cases} \tag{1.24}$$

où $\Omega^0 = \mathbb{R}^d - \Omega_1^0$. Et on doit vérifier la relation de compatibilité sur les contraintes

$$\begin{array}{c} [\sigma.n]_{métal}^{air} = W L_\epsilon^2 C_{a,m}^0 n \\ \sigma_0^{ext}.n - \sigma_0^{int}.n = W L_\epsilon^2 C_{a,m}^0 n. \end{array} \tag{1.25}$$

Remarques :

- A l'ordre 0, rien ne bouge.
- Comme dans le cas de la frontière fixe, le problème (P_{ext}) est surdéterminé et n'admet pas de solution de manière générale. De plus, il est peu probable de vérifier la relation sur les contraintes. C'est la condition de transmission qui rend le système surdéterminé.

Nous sommes en présence d'un problème de perturbation singulière, il existe donc une couche limite dans Ω_1^0 au voisinage de $\partial\Omega_1^0$.

Pour trouver un développement uniforme de B_ϵ, on peut utiliser **la technique des développements raccordés**([16]), pour cela il faudra chercher p_ϵ,u_ϵ et B_ϵ sous la forme

$$B_\epsilon(x,t) = \begin{pmatrix} B_0^{ext} \\ B_0^{int} \end{pmatrix} + \begin{pmatrix} 0 \\ B_0^{cl}(\gamma_1,\gamma_2,n,\epsilon,t) \end{pmatrix} + \epsilon B_1(x,t)$$
$$u_\epsilon(x,t) = u_0(x,t) + u^{cl}(\gamma_1,\gamma_2,n,\epsilon,t) + u_1$$
$$p_\epsilon(x,t) = p_0(x,t) + \epsilon u_1,$$

où (γ_1,γ_2) représente une paramétrisation normale de l'interface $\partial\Omega_1^0$.

L'introduction des termes de couche limite B^{cl},u^{cl}, va nous permettre de construire une approximation uniforme de B_ϵ,u_ϵ,p_ϵ, dans Ω_1^0. Le tenseur des contraintes va se décomposer de la manière suivante

$$\sigma = \sigma_0 + \sigma^{cl} + \epsilon\sigma_1.$$

- *Construction hors de la couche limite :* Les termes $B_0 = (B_0^{ext},B_0^{int}),u_0$, des développements de B_ϵ,u_ϵ, vérifient les équations du systèmes, (P_{ext}), *hors de la couche limite* ([16]).

  Proposition 3 *En première approximation, ordre 0, nous obtenons :*

  1. *la frontière du métal est fixe, et il n'y a pas d'interaction entre le fluide et le champ magnétique,*
  2. *le champ magnétique et les vitesses dans le métal sont donnés par les conditions initiales (ϕ_b^{int},ϕ_u) et la pression va s'ajuster pour vérifier la relation sur les contraintes.*
  3. *le champ magnétique à l'extérieur du métal est solution de*

$$\begin{cases} divB_0^{ext} = 0 \\ rotH_0^{ext} = J(x,t) \\ B_0^{ext} = \chi H_0^{ext} \\ B_0^{ext}.n = B_0^{int}.n = \phi_b^{int}.n \\ \lim\limits_{\|x\|\to+\infty} \|B_0^{ext}(x)\| = 0. \end{cases} \tag{1.26}$$

- *Calcul des fonctions provenant de la couche limite*: nous nous plaçons dans une approche bidimensionnelle et nous supposons que $\partial\Omega_1^0$ admet une paramétrisation normale.

  Soit $M \in \Omega_1$ près de $\partial\Omega_1$, il s'écrit, en coordonnées locales sous la forme M(s,r)=P(s)-rn(s) où P(s) est la projection orthogonale de M sur $\partial\Omega_1$.

  Nous avons le résultat suivant ([16]) :

  Proposition 4 *L'approximation à l'ordre 0 du champ magnétique, $B^{cl} = (B_s^0,B_n^0)$,*

et de la vitesse $u^{cl} = (u_s^0, u_n^0)$ *dans la couche limite est solution de*

$$\begin{cases} B_n^0 = 0 \\ \dfrac{\partial B_s^0}{\partial t} - \dfrac{\partial^2 B_s^0}{\partial r^2} = 0 \\ B_s^0(s,0,t) = \tau.(\chi B_0^{ext} - B_0^{int}) \\ \lim\limits_{r \to +\infty} B_s^0 = 0 \\ B(s,r,0) \quad \textit{donnée,} \end{cases} \tag{1.27}$$

$$\begin{cases} u_n^0 = 0 \\ \dfrac{\partial u_s^0}{\partial t} - \dfrac{E_m}{\chi_1}\left(\dfrac{R_{em}}{H_a}\right)^2 \dfrac{\partial^2 u_s^0}{\partial r^2} = 0 \\ \dfrac{\partial u_s^0}{\partial r} = -\dfrac{\partial}{\partial r}(u_0.\tau) \\ \lim\limits_{r \to +\infty} u_s^0 = 0 \\ u(s,r,0) \quad \textit{donnée} \end{cases} \tag{1.28}$$

où $u_0, B_0^{ext}, B_0^{int}$ *sont donnés par la proposition 3.*

Conclusion

A l'ordre du temps magnétique, tout est figé dans les conducteurs solides ou liquides excepté dans la zone de couche limite à la surface du conducteur. Cette zone est de très faible épaisseur dans le cas des hautes fréquences (inversement proportionnelle à la racine carrée de la fréquence du courant). Les méthodes de perturbations singulières permettent de construire, à l'ordre 0, des modèles où la peau magnétique est prise en compte comme un courant magnétique sur la surface du conducteur.

La faiblesse de cette approche (observation à l'échelle du temps magnétique) est qu'on ne peut suivre l'évolution de la forme du métal liquide. Une approche classique pour observer cette évolution à hautes fréquences est d'effectuer une approximation harmonique pour l'induction magnétique ([9],[44],[66]) et de considérer uniquement les équations de Navier-Stokes stationnaires avec des forces de Lorentz moyennes. Mais ce modèle interdit l'observation des phases de transition.

Pour la mise en place d'un modèle simplifié, on supposera l'existence d'une décomposition de l'induction magnétique en une partie oscillatoire (associée au temps court) et une partie non oscillatoire (associée à des temps plus long) :

$$\begin{aligned} j(x,t) &= \underline{j}(x,t)e^{i\omega t} \\ B(x,t) &= \underline{B}(x,t)e^{i\omega t}. \end{aligned} \tag{1.29}$$

Si l'on suppose que la densité de courant reste oscillatoire, alors la linéarité des équations de Maxwell permet d'éliminer la partie oscillatoire. Ceci permet de se ramener à des temps plus longs, d'observer l'évolution de la forme du conducteur liquide et si nécessaire les phases de transition. Si on est interessé par l'évolution du système jusqu'à un équilibre, on peut également utiliser les échelles à pas multiples ([17]).

1.3.3 Echelle à pas multiples

Dans de nombreuses applications, le temps magnétique est de l'ordre de 10^{-4} et le temps de diffusion est de l'ordre de la seconde. L'idée de la méthode à pas multiple est d' introduire une expression de l'échelle du temps long τ, sous la forme $\tau = \gamma(\epsilon)t$ où t est le temps court (le temps des pulsations magnétiques dans notre problème). Chaque paramètre du problème étudié est alors fonction de trois variables indépendantes **x**, τ, **t** et se décompose comme la somme d'un terme moyen en temps et d'un terme oscillant.

Par exemple, le champ magnétique d'induction est décomposé de la façon suivante:

$$B(x,\tau,t) = \widetilde{B}(x,\tau) + b(x,\tau,t)$$

où

$$\widetilde{B}(x,\tau) = \frac{1}{2\pi}\int_0^{2\pi} B(x,\tau,t)dt$$

b est 2π-périodique avec $\widetilde{b} = 0$.

Avec cette décomposition, nous obtenons deux systèmes non couplés, l'un satisfait par les valeurs moyennes, le second par les termes oscillants où seul le système associé au terme oscillant possède un petit paramètre. L'analyse asymptotique ne se fait que sur ce terme.

Dans le cas des conducteurs solides

On n'a pas de frontière libre et le système est linéaire. On observe les résultats suivants([18])

- A l'intérieur du conducteur, le terme oscillant du champ d'induction est uniquement concentré dans la couche limite d'épaisseur $\sqrt{\epsilon}$. Aussi, toutes les observations faites à l'extérieur de cette couche limite sont dues au terme moyen du champ d'induction.
- Dans le cas d'un courant induit harmonique et pour des temps longs, le comportement du champ magnétique d'induction est donné par son terme oscillant. La partie moyenne ne sert qu'à "relaxer" les conditions initiales. Cela peut-être vu comme une justification de l'approximation harmonique du champ d'induction.
- On obtient une expression explicite de l'induction magnétique au premier ordre. Ce qui permet de construire facilement la force de Joule dans les problèmes "thermal heating" et la force de Lorentz dans le cas de problème de lévitation.

Dans le cas des conducteurs liquides

Nous avons un problème à frontière libre et les systèmes non linéaires de Navier-Stokes et de Maxwell sont couplés. Pour les conducteurs liquides, la forme inconnue du conducteur dépend des deux paramètres de temps τ et t.

Le terme de premier ordre de la frontière libre ne dépend pas de t (temps court), ce qui permet d'intégrer comme dans le cas solide et d'obtenir un système couplé sur les composantes moyennes. Dans ce système, les composantes oscillantes interviennent dans le terme de couplage.

Chapitre 2

Vers l'écriture simplifiée de notre modèle

2.1 Introduction

Le principal objectif de ce chapitre est d'établir une écriture simplifiée d'un modèle numérique de notre problème physique. Cette écriture va nous permettre de mettre en évidence un type de problème à perturbation singulière à résoudre et de construire un algorithme de résolution adapté.

Dans le chapitre précédent, l'examen du problème physique a mis en évidence l'existence de deux temps caractéristiques, un temps court qui est le temps magnétique donné par la fréquence des pulsations magnétiques et un temps long qui correspond au temps de diffusion du champ d'induction dans le métal. Ce dernier ne dépend pas de la fréquence du courant imposé dans les conducteurs, mais des caractéristiques du conducteur lui-même. A l'échelle du temps magnétique, on observe l'aspect magnétique, à l'échelle du temps de diffusion, on observe l'évolution de la forme du conducteur liquide.

L'étude asymptotique a également montré que dans le cas d'un conducteur solide, comme dans le cas d'un conducteur liquide, tout se passe à l'intérieur d'une zone appelée **couche limite** dont l'épaisseur est inversement proportionnelle à la racine carrée de la fréquence du courant dans les inducteurs.

Dans la mise en place du modèle simplifié, nous ferons le choix d'un modèle nous permettant d'observer l'aspect magnétique tout en conservant la possibilité d'observer l'évolution de la frontière du conducteur liquide. L'adimensionalisation en temps du modèle de départ se fera par le temps des pulsations magnétiques ($\frac{1}{\omega}$) et nous supposerons l'existence d'une décomposition de l'induction magnétique en une partie oscillatoire (associée au temps des pulsations magnétiques) et une partie non oscillatoire (associée à des temps plus long) du type :

$$\underline{B}(x,t) = B(x,t)e^{it}. \tag{2.1}$$

L'élimination de la partie oscillatoire en reportant dans les équations de Maxwell va nous permettre de travailler sur des temps plus longs.

2.2 Équations de Maxwell

On se place dans le cas bidimensionnel et on étudie l'aspect magnétique. La mise adimensionnelle en temps se fait donc avec $\frac{1}{\omega_0}$ où ω_0 est la fréquence du courant imposé dans les inducteurs. Le domaine Ω du conducteur est un ouvert borné simplement connexe.

Soit $J_d(x,t)$, le courant dans les inducteurs et $B_d(x,t)$, l'induction magnétique, avec

$$\begin{array}{rcl} J_d(x,t) & = & \mathcal{R}e(\underline{J}(x,t)), \\ B_d(x,t) & = & \mathcal{R}e(\underline{B}(x,t)). \end{array} \tag{2.2}$$

On suppose que $\underline{J}$ et $\underline{B}$ se décompose en une partie oscillatoire associée au temps court et une partie non oscillatoire associée au temps long, sous la forme :

$$\begin{array}{rcl} \underline{J}(x,t) & = & J(x,t)\, e^{it}, \\ \underline{B}(x,t) & = & B(x,t)\, e^{it}, \end{array} \tag{2.3}$$

avec $J(x,t)$ et $B(x,t)$ complexes. Cette décomposition va nous permettre d'éliminer la partie oscillatoire associée au temps court dans les équations de Maxwell. On note $\Omega_1(t)$ et $\Omega(t)$ respectivement le domaine du conducteur et le domaine extérieur, au temps t .

En reportant dans (1.6, 1.10, 1.12), cela revient à résoudre le problème suivant :

o

$$\left\{ \begin{array}{ll} \dfrac{\partial \underline{B}}{\partial t} + \epsilon^2 Rotrot\underline{B} = \epsilon^2 R_{em} rot(u \wedge \underline{B}) & \text{dans} \quad \Omega_1(t) \\ rot\underline{B} = J & \text{dans} \quad \Omega(t) \\ div\underline{B} = 0 & \text{dans} \quad \mathbb{R}^2 \\ \underline{B} = \chi \underline{H} & \text{dans} \quad \mathbb{R}^2, \end{array} \right. \tag{2.4}$$

o les conditions limites (On a supposé que $\mu_0 = \mu_1$)

$$\left\{ \begin{array}{ll} [\underline{B}.n] = 0 & \text{dans} \quad \partial\Omega_1(t) \\ [\underline{H} \wedge n] = 0 & \text{dans} \quad \partial\Omega_1(t) \\ \lim\limits_{\|x\| \to +\infty} \|\underline{B}(x)\| = 0, & \end{array} \right. \tag{2.5}$$

o avec les hypothèses :

$$\left\{ \begin{array}{l} div\underline{J} = 0 \\ \displaystyle\int_\Omega \underline{J} = a \qquad \text{donné} \quad a \in \mathbb{C}. \end{array} \right.$$

Si on note :

$$\underline{B} = B(x,t)\, e^{it} = (B^r(x,t) + i.B^i(x,t))\, e^{it},$$
$$\underline{J} = J(x,t)\, e^{it} = (J^r(x,t) + i.J^i(x,t))\, e^{it},$$

on a les deux systèmes suivants à résoudre :

$$\begin{cases} \dfrac{\partial B^r}{\partial t} + \epsilon^2 RotrotB^r = \epsilon^2 R_{em} rot(u \wedge B^r) + B^i & \text{dans} \quad \Omega_1(t) \\ rotB^r = J & \text{dans} \quad \Omega(t) \\ divB^r = 0 & \text{dans} \quad \mathbb{R}^2 \\ B^r = \chi H^r & \text{dans} \quad \mathbb{R}^2 \\ [B^r.n] = 0 \quad \text{et} \quad [H^r \wedge n] = 0 & \text{dans} \quad \partial\Omega_1(t) \\ \lim\limits_{\|x\| \to +\infty} \|B^r(x)\| = 0 \end{cases} \tag{2.6}$$

avec les hypothèses

$$divJ^r = 0 \quad \text{et} \quad \int_\Omega J^r = a^r \qquad \text{donné} \quad a^r \in \mathbb{R}, \tag{2.7}$$

$$\begin{cases} \dfrac{\partial B^i}{\partial t} + \epsilon^2 RotrotB^i = \epsilon^2 R_{em} rot(u \wedge B^i) - B^r & \text{dans} \quad \Omega_1(t) \\ rotB^i = \mu_0 J^i & \text{dans} \quad \Omega(t) \\ divB^i = 0 & \text{dans} \quad \mathbb{R}^2 \\ B^i = \chi H^i & \text{dans} \quad \mathbb{R}^2 \\ [B^i.n] = 0 \quad \text{et} \quad [H^i \wedge n] = 0 & \text{dans} \quad \partial\Omega_1(t) \\ \lim\limits_{\|x\| \to +\infty} \|B^i(x)\| = 0 \end{cases} \tag{2.8}$$

avec les hypothèses

$$divJ^i = 0 \quad \text{et} \quad \int_\Omega J^i = a^i \qquad \text{donné} \quad a^i \in \mathbb{R}. \tag{2.9}$$

On ne conserve que la partie non oscillatoire du système de départ et les deux systèmes sont couplés par les termes B^i et B^r.

2.2.1 Formulation potentielle des équations de Maxwell

On note $(\mathcal{P}_{Maxw})$ le système défini par (2.6),(2.8) et les hypothèses (2.7),(2.9). $(\mathcal{P}_{Maxw})$ est un problème bien posé. On note B la solution de $(\mathcal{P}_{Maxw})$, alors :

Proposition : Soit $\Omega_1(t)$ un ouvert borné simplement connexe, il existe $\phi = \phi^r + i\phi^i$ unique définie sur $\mathbb{C}^2$ à valeurs dans $\mathbb{C}$ telle que : $B = Rot\phi$ et ϕ est solution du système

$$\begin{cases} \dfrac{\partial \phi^r}{\partial t} - \epsilon^2 \Delta\phi^r + \epsilon^2 R_{em} u.\nabla\phi^r = C^r(t) + \phi^i & \text{dans} \quad \Omega_1(t) \\ -\Delta\phi^r = J^r & \text{dans} \quad \Omega(t) \\ [\phi^r] = 0 \quad \text{et} \quad [\dfrac{\partial \phi^r}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1(t) \\ \lim\limits_{\|x\| \to +\infty} \phi^r(x) = 0 \end{cases} \tag{2.10}$$

$$\begin{cases} \dfrac{\partial \phi^i}{\partial t} - \epsilon^2 \Delta \phi^i + \epsilon^2 R_{em} u.\nabla \phi^i = C^i(t) + \phi^r & \text{dans} \quad \Omega_1(t) \\ -\Delta \phi^i = \mu J^i & \text{dans} \quad \Omega(t) \\ [\phi^i] = 0 \quad \text{et} \quad [\dfrac{\partial \phi^i}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1(t) \\ \lim\limits_{\|x\| \to +\infty} \phi^i(x) = 0 \end{cases} \tag{2.11}$$

avec $C^r = -\dfrac{\epsilon^2 a^r}{\text{mes}(\Omega_1(t))}$ et $C^i = -\dfrac{\epsilon^2 a^i}{\text{mes}(\Omega_1(t))}$.

Preuve:

- $\Omega_1(t)$ est un ouvert borné simplement connexe, la condition divB=0 sur $\mathbb{R}^2$ implique l'existence d'une fonction ϕ (fonction de courant) tel que $B = \text{Rot}\phi$ $(B_x = \dfrac{\partial \phi}{\partial y}$ et $B_y = \dfrac{\partial \phi}{\partial x})$.

  ϕ est déterminé à une constante près. On choisit ϕ tel que $\displaystyle\int_{\partial\Omega_1(t)} \phi(x,t) = 0$.

  On note $B^r(x,t) = \text{Rot}\phi^r(x,t)$ et $B^i(x,t) = \text{Rot}\phi^i(x,t)$.
- En remplaçant B par $\text{Rot}\phi$ dans (2.6),(2.8) on a

$$\begin{aligned} &\frac{\partial \phi^r}{\partial t} - \epsilon^2 \Delta^r + \epsilon^2 R_{em} u.\nabla \phi^r - \phi^i = C^r(t) \\ &\frac{\partial \phi^i}{\partial t} - \epsilon^2 \Delta^i + \epsilon^2 R_{em} u.\nabla \phi^i + \phi^r = C^i(t). \end{aligned} \tag{2.12}$$

- Déterminons $C^r(t)$ et $C^i(t)$

  On a

  $$\int_{\Omega_1(t)} [\frac{\partial \phi^r}{\partial t} + \epsilon R_{em} u.\nabla \phi^r] = \frac{\partial}{\partial t} \int_{\Omega_1(t)} \phi^r(t) = 0,$$

  $$\int_{\Omega_1(t)} \phi^i = 0,$$

  $$\int_{\Omega_1(t)} \Delta \phi^r = \int_{\Omega(t)} J^r = a^r.$$

  En reportant dans (2.12), on obtient

  $$C^r(t) = -\frac{\epsilon a^r}{\text{mes}(\Omega_1(t))}.$$

  On utilise le même raisonnement pour $C^i(t)$.

■

2.2.2 Discrétisation en temps

Pour la résolution de (2.6),(2.8), on utilise **un schéma semi-implicite** pour la discrétisation en temps, un schéma d'*Euler avancé* pour le terme de convection, un schéma de

Crank-Nilcolson pour le terme de diffusion. On note Ω_1^n et Ω^n respectivement le domaine du conducteur et le domaine extérieur, au temps $n\,\Delta t$ et $\partial\Omega_1^n$ la frontière entre ces deux domaines au temps $n\,\Delta t$. On obtient :

Partie réelle :

$$\begin{cases} \dfrac{\phi^{r,n+1}-\phi^{r,n}}{\Delta t} - \epsilon^2\theta\Delta\phi^{r,n+1} = G^{r,n} + \phi^{i,n} & \text{dans} \quad \Omega_1^n \\ -\Delta\phi^{r,n+1} = J^{r,n+1} & \text{dans} \quad \Omega^n \\ [\phi^{r,n+1}] = 0 \quad \text{et} \quad [\dfrac{\partial\phi^{r,n+1}}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1^n \end{cases} \tag{2.13}$$

avec $G^{r,n} = \epsilon^2(1-\theta)\Delta\phi^{r,n} - \epsilon^2 R_{em} u^{n+1}.\nabla\phi^{r,n} + C^{r,n}$.

Partie Imaginaire :

$$\begin{cases} \dfrac{\phi^{i,n+1}-\phi^{i,n}}{\Delta t} - \epsilon^2\theta\Delta\phi^{i,n+1} = G^{i,n} - \phi^{r,n} & \text{dans} \quad \Omega_1^n \\ -\Delta\phi^{i,n+1} = J^{i,n+1} & \text{dans} \quad \Omega^n \\ [\phi^{i,n+1}] = 0 \quad \text{et} \quad [\dfrac{\partial\phi^{i,n+1}}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1^n \end{cases} \tag{2.14}$$

avec $G^{i,n} = \epsilon^2(1-\theta)\Delta\phi^{i,n} - \epsilon^2 R_{em} u^{n+1}.\nabla\phi^{i,n} + C^{i,n}$ où θ est le paramètre du schéma de discrétisation en temps de Crank-Nilcolson ($\theta \in [0,1]$).

Remarques :

· On a vu précédemment que les deux systèmes *"partie imaginaire"* et *" partie réelle"* sont couplés. On peut envisager deux méthodes pour résoudre ce couplage.

- On rend explicite le terme de couplage dans la discrétisation en temps. C'est cette méthode qui a été choisie pour le modèle simplifié ci-dessus. Elle permet également de résoudre en parallèle les systèmes *"partie imaginaire"* et *" partie réelle"*.
- La deuxième méthode consiste à résoudre le système couplé à deux inconnues ϕ^r et ϕ^i par un gradient conjugué.

2.2.3 Écriture simplifiée du modèle

Nous venons de voir que pour chaque itération en temps, on a le modèle suivant :

Partie réelle :

$$\begin{cases} \phi^{r,n+1} - \alpha\Delta\phi^{r,n+1} = \phi^{r,n} + \Delta t\,(G^{r,n} + \phi^{i,n}) & \text{dans} \quad \Omega_1^n \\ -\Delta\phi^{r,n+1} = J^{r,n+1} & \text{dans} \quad \Omega^n \\ [\phi^{r,n+1}] = 0 \quad \text{et} \quad [\dfrac{\partial\phi^{r,n+1}}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1^n \end{cases} \tag{2.15}$$

avec $\alpha = \epsilon^2\theta\Delta t$,

Partie Imaginaire :

$$\begin{cases} \phi^{i,n+1} - \alpha\Delta\phi^{i,n+1} = \phi^{i,n} + \Delta t\,(G^{i,n} - \phi^{r,n}) & \text{dans} \quad \Omega_1^n \\ -\Delta\phi^{i,n+1} = J^{i,n+1} & \text{dans} \quad \Omega^n \\ [\phi^{i,n+1}] = 0 \quad \text{et} \quad [\dfrac{\partial\phi^{i,n+1}}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1^n \end{cases} \tag{2.16}$$

où α est un petit paramètre des deux systèmes.

Cela revient à résoudre **le problème de transmission** suivant :

$$\begin{cases} \phi - \alpha\Delta\phi = \mathcal{F} & \text{dans} \quad \Omega_1 \\ -\Delta\phi = J & \text{dans} \quad \Omega \\ [\phi] = 0 \quad \text{et} \quad [\dfrac{\partial\phi}{\partial n}] = 0 & \text{dans} \quad \partial\Omega_1 \\ \text{décroissance à l'infini.} \end{cases} \tag{2.17}$$

On notera :

$$E(\phi^{n+1},\partial\Omega_1^{n+1}) = F_m(\phi^n,u^{n+1},\partial\Omega_1^{n+1}).$$

2.2.4 Résolution

L'étude effectuée dans le chapitre précédent, nous a montré que lorsque la fréquence augmente, on est en présence d'un problème à perturbation singulière où tout se passe dans la couche limite. Notre problème présente deux caractéristiques :

- une transmission à la frontière,
- une couche limite au niveau de la frontière du conducteur.

Dans le chapitre suivant, nous ferons l'étude d'une **méthode de décomposition de domaine sans recouvrement** adaptée à ces deux caractéristiques.

2.3 Equations de Navier-Stokes

2.3.1 Formulation vitesse - pression

Dans le problème traité, on est dans le cas d'un écoulement incompressible d'un fluide newtonien homogène soumis à des forces f et occupant un domaine $\Omega_1(t)$, ouvert borné

qu'on suppose simplement connexe de $\mathbb{R}^2$ et de frontière suffisamment régulière.

Les équations de **Navier-Stokes** formulées en variables primitives, vitesse u(x,t) et pression p(x,t) de notre problème (1.8, 1.9,1.10) s'écrivent :

$$(\mathcal{P}_{NS}) \begin{cases} \dfrac{1}{\epsilon^2 R_{em}} \dfrac{\partial u}{\partial t} + (u.\nabla)u + \nabla p - Re^{-1}\Delta u = f_L & \text{dans} \quad \Omega_1(t) \\ divu = 0 & \text{dans} \quad \Omega_1(t) \\ u.n = v_n = 0 & \text{sur} \quad \partial\Omega_1(t) \\ (\nabla u + \nabla u^T).n.\tau = 0 & \text{sur} \quad \partial\Omega_1(t) \\ u(x,0) = u_0(x) \quad \text{donné.} \end{cases} \tag{2.18}$$

On a noté f_L la force de Lorentz avec $f_L = \chi^{-1} rot(B) \wedge b$ et v_n la normale de la vitesse particulaire sur $\partial\Omega_1(t)$. On cherche un état stationnaire, on prendra v_n=0. Le problème $(\mathcal{P}_{NS})$ est bien posé dès que les données initiales sont suffisamment régulières (voir entre autres J.L. Lions [45], R. Teman [70]).

2.3.2 Formulations vitesse - tourbillon

La formulation *vitesse - tourbillon* (u,ω) des équations de Navier-Stokes incompressible dérive directement de leur formulation en *vitesse-pression* par l'introduction du vecteur tourbillon en appliquant l'opérateur **rotationnel** à l'équation de transport de la vitesse.

On introduit l'opérateur **K** (**K** est le noyau de Biot et Savart associé à Ω) définit par :

$$\begin{aligned} Kf(t,z) &= Rot \int_\Omega G(z,z')f(t,z')dz' \\ &= \begin{cases} \dfrac{\partial}{\partial y}\displaystyle\int_\Omega G(z,z')f(t,z')dz' \\ -\dfrac{\partial}{\partial x}\displaystyle\int_\Omega G(z,z')f(t,z')dz' \end{cases} \end{aligned}$$

où z=(x,y) et G est *la fonction de Green* de Δ sur Ω.
On rappelle que Ω est un ouvert borné, simplement connexe de $\mathbb{R}^2$. On a le lemme suivant ([7]) :

Lemme 1 *Soit $u : \bar{\Omega} \longrightarrow \mathbb{R}^2$ étant une fonction régulière. Alors (i) et (ii) sont équivalents :*

(i) div u=0 dans $\bar{\Omega}$, u.n=0 sur $\partial\Omega$
(ii) Il existe $\omega : \bar{\Omega} \longrightarrow \mathbb{R}$ tel que u=Kω.

Preuve:
$(i) \rightarrow (ii)$ On pose $\omega = rotu$. Puisque Ω est un ouvert simplement connexe, la condition divu=0 implique l'existence d'une fonction ψ (fonction de courant) telle que $u = Rot\psi$. Si on prend une paramétrisation de $\partial\Omega$, on montre facilement :

$$u.n(t,z) = 0 \quad \forall t \geq 0 \quad \forall z \in \partial\Omega \iff \psi(t,z) = c \quad \forall t \geq 0 \quad \forall z \in \partial\Omega. \tag{2.19}$$

Ainsi ψ est définie à une constante près et on peut choisit c=0.

Puisque ω=rotu, on a de façon immédiate $\omega = \Delta\psi$. Puisque ψ est nulle sur $\partial\Omega$, elle peut être exprimée en fonction de ω et de la fonction de Green.

$$\psi(t,z) = \int_\Omega G(z,z')\omega(z',t)dz'. \tag{2.20}$$

Ce qui nous donne $\omega = Ku$.

$(ii) \rightarrow (i)$ On suppose que $u = K\omega$, on a par définition $u = Rot\,\psi$ avec ψ défini par (2.20). Par conséquent $div\,u = 0$ sur $[0,\infty[\times\Omega$ et $\psi(x,t) = 0\ \forall\ (t,z)$ sur $[0,\infty[\times\partial\Omega$. On déduit, à l'aide de (2.19) que u.n=0

■

En utilisant le lemme 1, on obtient facilement le théorème suivant :

Théorème 1 *Le problème $(\mathcal{P}_{NS})$ + $(\omega = rotu)$ est équivalent au problème $(\mathcal{P}'_{NS})$ suivant :*

$$(\mathcal{P}'_{NS}) \begin{cases} \dfrac{1}{\epsilon^2 R_{em}}\dfrac{\partial\omega}{\partial t} + div(u.\omega) - Re^{-1}\Delta\omega = rotf & dans \quad \Omega_1(t) \\ \omega = rotu & dans \quad \Omega_1(t) \\ (\nabla u + \nabla u^T).n.\tau = 0 & sur \quad \partial\Omega_1(t) \\ u(x,0) = u_0(x) \quad donné & \\ \omega_0(x) = rotu_0 \quad donné. & \end{cases} \tag{2.21}$$

Preuve: $\Omega_1(t)$ est un ouvert borné, supposé simplement connexe de $\mathbb{R}^2$.

(i) L'implication est immédiate en appliquant le rotationnel à l'équation de convection-diffusion de la formulation vitesse-pression.

(ii) Pour toutes fonctions régulières u et q de $\mathbb{R}^2$ dans $\mathbb{R}^2$, nous avons :

$$rot(\nabla q) = 0$$
$$Rot((u.\nabla)u) = \omega.\nabla(rotu) + (divu)rotu.$$

Si ω est solution de $(\mathcal{P}'_{NS})$, d'après le lemme(1), si on pose $u = K\omega$, on a

$$\text{divu=0 et u.n(t,x)=0 } \forall t \geq 0 \quad \text{et} \quad x \in \partial\Omega_1.$$

D'autre part,

$$Rot(\frac{1}{\epsilon^2 R_{em}}\frac{\partial u}{\partial t} + (u.\nabla)u - Re^{-1}\Delta u - f) = 0.$$

Il existe une fonction p telle que

$$\frac{1}{\epsilon^2 R_{em}}\frac{\partial u}{\partial t} + (u.\nabla)u - Re^{-1}\Delta u - f = -\nabla p.$$

u est solution de $(\mathcal{P}_{NS})$.

■

Remarque : L'équation sur la vitesse du problème $(\mathcal{P}'_{NS})$ est du premier ordre, c'est pourquoi on se ramène en général à un problème $(\mathcal{P}''_{NS})$ équivalent qui fait intervenir une équation du second ordre sur la vitesse.

2.3.3 Formulation ω-ψ des équations de Navier-Stokes

Pour satisfaire exactement la condition d'incompressibilité (divu=0), on travaillera directement avec **une fonction de courant** ψ ($u = Rot\,\psi$). Pour cela, on utilisera les résultats du lemme 1.

Théorème 2 *Le problème* $(\mathcal{P}'_{NS})$ *est équivalent au problème* $(\mathcal{P}''_{NS})$ *suivant:*

$$(\mathcal{P}''_{NS})\begin{cases} \dfrac{1}{\epsilon^2 R_{em}}\dfrac{\partial\omega}{\partial t}+\dfrac{\partial\omega}{\partial x}.\dfrac{\partial\psi}{\partial y}-\dfrac{\partial\omega}{\partial y}.\dfrac{\partial\psi}{\partial x}-Re^{-1}\Delta\omega=rotf & dans \quad \Omega_1(t) \\ \omega=-\Delta\psi & dans \quad \Omega_1(t) \\ \psi=0 & sur \quad \partial\Omega_1(t) \\ \omega=2\times\frac{\partial^2\psi}{\partial r^2} & sur \quad \partial\Omega_1(t) \\ \omega_0=rotu_0 \quad u_0\ donné & sur \quad \Omega_1(t). \end{cases} \tag{2.22}$$

Preuve:

- L'équivalence est immédiate pour les équations de convection-diffusion et de diffusion sur Ω.
- **conditions au bord**

  (i) condition de non cisaillement

$$(\nabla u+\nabla^T u)n.\tau=0 \quad \text{sur} \quad \partial\Omega_1(t).$$

  - On a dans le repère de la surface([6])

$$\sigma=(\nabla u+\nabla u^T)=\begin{pmatrix} e_{rr} & e_{r\theta} \\ e_{\theta r} & e_{\theta\theta}\end{pmatrix}$$

avec

$$\begin{aligned} e_{rr} &= \frac{\partial u_r}{\partial r} \\ e_{\theta\theta} &= \frac{1}{r}\frac{\partial u_\theta}{\partial r}+\frac{u_r}{r} \\ e_{r\theta} &= \frac{r}{2}\frac{\partial}{\partial r}(\frac{u_\theta}{r})+\frac{1}{2r}\frac{\partial u_r}{\partial\theta}. \end{aligned}$$

On exprime alors la condition de glissement par

$$\sigma n.\tau=e_{\theta r}=r\frac{\partial}{\partial r}(\frac{1}{r}\frac{\partial\psi}{\partial r})-\frac{1}{r^2}\frac{\partial^2\psi}{\partial\theta^2}=0. \tag{2.23}$$

- $divu=\dfrac{1}{r}\dfrac{\partial(ru_r)}{\partial r}+\dfrac{1}{r}\dfrac{\partial u_\theta}{\partial\theta}$

$$divu=0\Longrightarrow u=\begin{pmatrix} u_r \\ u_\theta\end{pmatrix}=\begin{pmatrix} -\frac{1}{r}\frac{\partial\psi}{\partial\theta} \\ \frac{\partial\psi}{\partial r}\end{pmatrix} \tag{2.24}$$

-

$$\begin{aligned} \omega &= rotu \\ &= \frac{1}{r}\frac{\partial}{\partial r}(ru_\theta)-\frac{1}{r}\frac{\partial u_r}{\partial\theta}. \end{aligned} \tag{2.25}$$

En utilisant les égalités (2.24) et (2.25), on montre que la condition de non cisaillement (2.23) est équivalente à :

$$\omega = 2\frac{\partial^2 \psi}{\partial r^2}.$$

(ii) condition de glissement

$$u.n = \nabla\psi.\tau = \frac{\partial \psi}{\partial \tau}.$$

On a

$$\frac{\partial \psi}{\partial \tau} = 0.$$

La fonction ψ est constante sur la frontière, elle est déterminée à une constante près, on prendra $\psi = 0$.

C'est le lemme 1 qui nous permet de démontrer la réciproque. En effet, soit (ψ,ω) solution de $(\mathcal{P}''_{NS})$, on pose $u = Rot\,\psi$. D'après le lemme 1, on a :

$$u.n = 0 \quad \text{sur} \quad \partial\Omega \quad \text{et} \quad div\, u = 0 \quad \text{sur} \quad \bar{\Omega}$$

On a également $\omega = rot\, u$. On peut donc en déduire la condition de non cisaillement. ■

2.3.4 Discrétisation en temps

On est ramené au problème suivant :

$$(\mathcal{P}''_{NS}) \begin{cases} \dfrac{\partial \omega}{\partial t} + \epsilon^2 R_{em} J(\omega,\psi) - \epsilon^2 R_{em} Re^{-1} \Delta\omega = \epsilon^2 R_{em} rot f & \text{dans} \quad \Omega_1(t) \\ \omega = 2 \times \dfrac{\partial^2 \psi}{\partial r^2} & \text{sur} \quad \partial\Omega_1(t) \\ -\Delta\psi = \omega & \text{dans} \quad \Omega_1(t) \\ \psi = 0 & \text{sur} \quad \partial\Omega_1(t) \\ \omega_0 = rot u_0 \quad u_0 \text{ donné} & \text{sur} \quad \partial\Omega_1(0) \end{cases} \tag{2.26}$$

où J est le Jacobien de ω et ψ :

$$J(\omega,\psi) = \frac{\partial \omega}{\partial x}.\frac{\partial \psi}{\partial y} - \frac{\partial \omega}{\partial y}.\frac{\partial \psi}{\partial x}.$$

On utilise un schéma semi-implicite pour la discrétisation en temps :

- **Euler** avancé, totalement explicite pour le terme de convection $J(\omega,\psi)$,
- **Crank-Nicolson** de paramètre θ pour le terme de diffusion.

Le problème devient :

$$(\mathcal{P}''_{NS}) \begin{cases} \dfrac{\omega^{n+1} - \omega^n}{\Delta t} - \epsilon^2 \theta R_{em} Re^{-1} \Delta\omega^{n+1} = \mathcal{F}^n & \text{dans} \quad \Omega_1^n \\ \omega^{n+1} = 2 \times \dfrac{\partial^2 \psi^n}{\partial r^2} & \text{sur} \quad \partial\Omega_1^n \\ -\Delta\psi^{n+1} = \omega^{n+1} & \text{dans} \quad \Omega_1^n \\ \psi^{n+1} = 0 & \text{sur} \quad \partial\Omega_1^n \\ \omega_0 = rot u_0 \quad u_0 \text{ donné} & \text{sur} \quad \partial\Omega_1^0 \end{cases} \tag{2.27}$$

avec $\mathcal{F}^n = \epsilon^2 R_{em}((1-\theta)\Delta\omega^n + J(\omega^n,\psi^n) + Rotf^n$ et Ω_1^n, le domaine du conducteur au temps $n\,\Delta t$.

Remarque : On aurait pu utiliser les schémas de **Adams-Bashforth** pour le terme de convection([25]).

- d'ordre 2

$$U^{n+1} = U^n + \Delta t(\frac{3}{2}F^n - \frac{1}{2}F^{n-1}),$$

- d'ordre 3

$$U^{n+1} = U^n + \Delta t(\frac{22}{12}F^n - \frac{16}{12}F^{n-1} + \frac{15}{12}F^{n-2}).$$

La taille de la région de stabilité décroit avec l'ordre du schéma.

2.3.5 Écriture simplifiée du modèle simplifié

Nous avons vu précédemment que pour chaque itération en temps, le problème $(\mathcal{P}''_{NS})$ peut s'écrire :

$$(\mathcal{P}''_{NS})\begin{cases} \omega^{n+1} - \alpha\Delta\omega^{n+1} = \Delta t\,\mathcal{F}^n + \omega^n & \text{dans} \quad \Omega_1^n \\ \omega^{n+1} = 2 \times \dfrac{\partial^2\psi^n}{\partial r^2} & \text{sur} \quad \partial\Omega_1^n \\ -\Delta\psi^{n+1} = \omega^{n+1} & \text{dans} \quad \Omega_1^n \\ \psi^{n+1} = 0 & \text{sur} \quad \partial\Omega_1^n \\ \omega_0 = rotu_0 \quad u_0 \text{ donné} & \text{dans} \quad \Omega_1^0 \end{cases} \tag{2.28}$$

avec $\alpha = \epsilon^2\theta R_{em} Re^{-1}\Delta t$.

Nous sommes amené à résoudre **le problème couplé** suivant :

$$\begin{cases} \omega - \alpha\Delta\omega = \mathcal{F} & \text{dans} \quad \Omega_1 \\ \omega = 2 \times \dfrac{\partial^2\psi}{\partial r^2} & \text{sur} \quad \partial\Omega_1 \\ -\Delta\psi = \omega & \text{dans} \quad \Omega_1 \\ \psi = 0 & \text{sur} \quad \partial\Omega_1. \end{cases} \tag{2.29}$$

On notera

$$H(\psi^{n+1},\omega^{n+1},\partial\Omega_1^{n+1}) = F_L(\psi^n,\omega^n,\phi^{n+1},\partial\Omega_1^{n+1}).$$

Pour les hautes fréquences, on a α petit paramètre, on est en présence d'un problème à perturbation singulière. On étudiera dans le chapître suivant une méthode numérique adaptée à ce problème simplifié.

Remarque : On obtient un modèle simplifié avec le même type d'opérateur si on utilise les schémas d'Adams-Bashforth pour la discrétisation en temps.

2.4 Frontière libre

Les conditions qui doivent être satisfaites à la surface d'un liquide (ou d'un gaz) sont de deux types :

- **la condition cinématique** qui suppose qu'une particule de fluide située sur la surface libre a une vitesse égale à la vitesse de cette surface libre.

  Si l'interface métal-air est représentée par l'équation

  $$F(x,t) = 0,$$

  et si on note n(x,t) la normale à F en (x,t), on a vu dans la présentation physique du problème que l'on a

  $$u.\nabla F + \frac{\partial F}{\partial t} = 0 \quad \text{avec} \quad \nabla F = < \nabla F, n > n,$$

  soit encore :

  $$u.n = -\frac{1}{< \nabla F.n >}\frac{\partial F}{\partial t} \;=\; v_n$$

  où v_n est la vitesse normale de déplacement de l'interface. On cherche un état stationnaire, on prendra $v_n = 0$.
- **Une condition dynamique** qui traduit l'équilibre entre les contraintes et les forces de tensions superficielles ([19],[42]).

  $$[\sigma.n]^{air}_{métal}.\underline{n} = WL_e^2 C_{a,m} \quad \text{sur} \quad \partial\Omega_1,$$

  avec

  $$\begin{aligned} \underline{\sigma}_{air} &= -(p_a + \tfrac{1}{2}B^a.B^a) + B^a \otimes B^a \\ \underline{\sigma}_{métal} &= -(p + \frac{1}{2}B^m.B^m) + B^m \otimes B^m + R_e^{-1}(\nabla u + \nabla u^T) \end{aligned}$$

  et $B \otimes B = (B_i B_j)_{ij}$.

  On pose

  $$\begin{cases} P_m &= p + \dfrac{1}{2}B^m.B^m \\ P_A &= p_a + \dfrac{1}{2}B^a.B^a. \end{cases}$$

  On a

  $$[\sigma.n]^{air}_{métal} = (P_m - P_A)\underline{n} + R_e^{-1}(\nabla u + \nabla u^T)\underline{n}$$

  $$P + R_e^{-1}(\nabla u + \nabla u^T)\underline{n}.\underline{n} = WL_e^2 C_{a,m}.$$

2.4.1 Détermination de la courbure

On exprime (1.4) dans le repère (n,τ) par :

$$P + R_e^{-1}\frac{\partial u_n}{\partial n} = WL_e^2 C_{a,m}.$$

En dérivant par rapport à τ et en utilisant la condition cinématique (u.n=0), on obtient

$$\frac{\partial P}{\partial \tau} = WL_e^2\frac{\partial C_{a,m}}{\partial \tau}.$$

La courbure est périodique, elle admet un développement en série de Fourier.

$$C_{a,m} = \sum_k b_k e^{ik\theta}$$

$$\frac{\partial C_{a,m}}{\partial \tau} = \sum_k ikb_k e^{ik\theta}$$

et

$$\frac{\partial P}{\partial \tau} = \nabla p \wedge \underline{n}.$$

UNIVERSITÉ NANCY 1 BIBLIOTHÈQUE SCIENCES

Calcul de ∇p

La pression p n'est pas une variable de la formulation choisie. Elle sera éliminée en utilisant l'équation de convection-diffusion de la formulation en variables primitives ([33])

$$\frac{1}{\epsilon^2 R_{em}}\frac{\partial u}{\partial t} + (u.\nabla)u + \nabla p + R_e^{-1}\Delta u = f_L.$$

En remplaçant u par son expression en fonction de ψ (fonction de courant) ou ω (vecteur tourbillon) on a :

$$\left\{\begin{array}{rcl} \frac{1}{\epsilon^2 R_{em}}\frac{\partial^2\psi}{\partial t\partial y} + (\frac{\partial\psi}{\partial y}\frac{\partial^2\psi}{\partial y\partial x} - \frac{\partial\psi}{\partial x}\frac{\partial^2\psi}{\partial^2 y}) + p_x - R_e^{-1}w_y &=& \frac{\partial f_L}{\partial x} \\ \frac{1}{\epsilon^2 R_{em}}\frac{\partial^2\psi}{\partial t\partial x} + (\frac{\partial\psi}{\partial x}\frac{\partial^2\psi}{\partial x\partial y} - \frac{\partial\psi}{\partial y}\frac{\partial^2\psi}{\partial^2 x}) + p_y - R_e^{-1}w_x &=& \frac{\partial f_L}{\partial y}. \end{array}\right. \tag{2.30}$$

De ces deux équations, on peut extraire ∇p.

Détermination de la frontière

Nous choisissons un système de coordonnées polaires dont l'origine x_0 est un point arbitraire de l'axe x. La paramétrisation d'un point M de $\partial\Omega_1$ est donnée par :

$$\begin{array}{rcl} x_1 &=& x_0 + r(\theta)\cos(\theta) \\ x_2 &=& r(\theta)\sin(\theta) \quad \text{avec} \quad \theta \in I = [0,\overline{\theta}] \end{array}$$

où $r(\theta)$ est la fonction inconnue. Pour ce système de coordonnées, la courbure C est donnée par ([9]) :

$$c(\theta) = \frac{r^2(\theta) + 2r'^2(\theta) + r(\theta)r''(\theta)}{(r^2(\theta) + r'(\theta)^2)^{\frac{3}{2}}}.$$

Si la fonction $r(\theta)$ avec $\theta \in I$ est connue, son graphe détermine $\partial\Omega_1$ et par la même occasion Ω_1. Le problème de frontière libre peut s'écrire sous la forme

$$F(\psi^{n+1},\omega^{n+1},\partial\Omega_1^{n+1}) = 0.$$

La dépendance en ψ et ω provient du terme du tenseur des contraintes.

2.5 Formulation simplifiée du modèle complet

Notre problème consiste à déterminer $(\psi,\omega,\phi,\partial\Omega_1)$ à chaque pas de temps. On a à résoudre :

- *Electromagnétisme :* $E(\phi^{n+1},\partial\Omega_1^{n+1}) = F_m(\phi^n,u^{n+1},\partial\Omega_1^{n+1})$,
- *Hydrodynamique :* $H(\psi^{n+1},\omega^{n+1},\partial\Omega^{n+1}) = F_L(\phi^{n+1},\psi^n,\omega^n,\partial\Omega_1^{n+1})$,
- *Frontière libre :* $F(\omega^{n+1},\psi^{n+1},\partial\Omega_1^{n+1}) = 0$.

On déduit u^{n+1} et p^{n+1}.

La procédure itétrative consiste à :

1. Initialiser $\partial\Omega_1^0$ (par la même occasion le domaine du métal liquide Ω_1^0) ainsi que u_0
2. Initialiser n (n=0)
3. Générer une discrétisation en espace de Ω_1^n
4. Résoudre le problème d'electromagnétisme :$E(\phi^{n+1},\partial\Omega_1^{n+1}) = F_m(\phi^n,u^n,\partial\Omega_1^{n+1})$
5. Résoudre le problème d'hydrodynamisme : $H(\psi^{n+1},\omega^{n+1},\partial\Omega^{n+1}) = F_L(\phi^{n+1},\psi^n,\omega^n,\partial\Omega_1^{n+1})$
6. Déterminer la frontière libre : $F(\omega^{n+1},\psi^{n+1},\partial\Omega_1^{n+1}) = 0$
7. Tester la convergence de la distance entre $\partial\Omega_1^n$ et $\partial\Omega_1^{n+1}$. Si la convergence n'est pas atteinte, aller à l'étape 3.

Dans le chapitre suivant, nous étudierons un algorithme adapté au phénomène de perturbation singulière figurant dans le problème d'électromagnétisme (2.17) et le problème d'hydrodynamisme (2.28).

Chapitre 3

A domain decomposition adapted to our problem

3.1 Introduction

In this chapter our goal is to study homogeneous and heterogeneous domain decomposition methods in order to give a good algorithm to resolve singular perturbation problems which we displayed prominently in our global physical problem. At the magnetic time, in high frequencies, all is fixed in the conductor except in the magnetic skin located near the conductor surface. The solutions for the singular perturbation problem typically display rapid variations across narrow region, the so-called boundary layer. It is then challenging to design a numerical scheme that gives an uniform approximation of the solution over this region.

Singular perturbation problems have been studied extensively in asymptotic analysis but the results of these studies have not yet been fully exploited to enhance numerical computations (see for example [24], [23], [58] and their references). Following the method in the papers [27] [28] [29] [30], we systematically use the concept of singular perturbation to design efficient and accurate domain decomposition solvers for these problems.

- The subdomains of computation correspond to the "*regular domains*" and the "*singular layers*" of the S.P.P,
- the position of the interfaces depends on the small parameter ϵ, and is determined in such a way that the truncation error is asymptotically of the same order in each subdomain,
- the aspect ratio of the mesh changes drastically from a regular subdomain to a singular layer subdomain.

In previously published work [28] [29], we investigated a Schwarz alternating procedure implemented in its most elementary form. It was shown that the rate of convergence of the iterative procedure is superlinear when the domain decomposition is properly designed. Superlinear rate of convergence means here that the global convergence rate depends on ϵ and goes to zero when $\epsilon \longrightarrow 0$.

This method can be used to solve the vorticity equation or magnetic field equation on the metal but we can't use it to solve the transmission problem that arises in the electromagnetic phenomenon. We had to study a domain decomposition method without recovering adapted to our electromagnetic transmission problem.

In this paper we prove similar properties of the F.Q algorithm and emphazise the impact of the choice of boundary conditions . We consider classical second order linear elliptic singular perturbation problem with zero order degeneracies [23] and a transmission problem that arises in electromagnetic theory. We give rigorous convergence and accuracy estimates in the maximum norm in the finite difference framework but we apply our results to enhance finite element computations.

In the following paragraph, we study in details homogeneous and heterogeneous domain decompositions in one space dimension (with or without transmission).

The asymptotic analysis of our transmission problem suggests that the computation domain should be split into three subdomains, using the intermediate domain to solve the boundary layer. Using at once (F.Q.) and Schwarz procedures, we compared three numeric schemes allowing to solve our electromagnetic transmission problem. The first and the second schemes use the (F.Q) procedure to solve the boundary layer and the transmission problem with different artificial boundary conditions at the conductor surface, whereas the third scheme use the (F.Q.) procedure to resolve the transmission problem and the Schwarz procedure to solve the boundary layer.

In the third paragraph, we generalize these results to a two dimensional space via a comparison lemma in the case of circular geometries. We implemented, with Matlab, the three numerical schemes using finite difference approximations. This implementation confirmed the theorical results and the seriousness of the choice of the boundary conditions at the conductor surface. At last, in the same context, we implemented these three schemes using the finite element approximations thanks to the Modulef library ([8]). The results obtained seem to indicate that these schemes could be extended to more general geometries.

We apply our main results in Sect 4 to enhance the computation of an elementary two dimensional singular perturbed transmission problem that arises in electromagnetic theory with *Modulef*. This chapter issued from a work in co-operation with Marc Garbey and Olivier Coulaud ([31]).

3.2 Boundary layers in a One Dimensional Space

3.2.1 Homogeneous Domain Decomposition

In this section, we consider a linear second-order singular perturbation problem of the following type:

$$\begin{cases} L_\epsilon \phi = -\epsilon \phi'' + \phi = F \ in \ \Omega = (0,1), \\ \phi(0) = \alpha_0 \ ; \ \phi(1) = \alpha_1. \end{cases} \tag{3.1}$$

Where ϵ is a small positive parameter in $]0,\epsilon_0]$ for some $\epsilon_0 > 0$. Problems of this type exhibit boundary layers usually at both ends of the interval. This trivial one dimensional problem will be used as an illustration for our method. In particular we will show how properly design the domain decomposition in order to get fast convergence for the Quarteroni-Funaro iterative solver with *no relaxation* parameter.

We restrict ourselves to the case of a single boundary layer in the neighborhood of 1. According to the asymptotic analysis, we should split the domain Ω into two subdomains $\Omega_{inner} = (a,1)$ and $\Omega_{outer} = (0,a)$ where $a > 0$. Ω_{inner} covers the boundary layer at 1 and Ω_{outer} covers the domain of validity of the regular approximation.

In order to easily get sharp estimates in the maximum norm, we are going to use the finite difference framework.

We keep the mesh regular in each subdomain and adapt the domain decomposition to the boundary layer stiffness. Let us denote h_1 (respt. h_2) the mesh size on Ω_{outer} (respt Ω_{inner}). Let us denote by L^{h_i}, $i = 1,2$ the discretized operator that corresponds to L_ϵ. We will also restrict ourselves to the case where we have the same asymptotic order of grid points in each subdomain, i.e

$$h_1 \approx \frac{h_2}{1-a} \approx \frac{1}{N}, \tag{3.2}$$

in order to balance the amount of work in each subdomain.

Dirichlet-Neumann scheme

We introduce the following iterative procedure [26] in order to solve(3.1)

$$\begin{cases} L^{h_1} \phi^p_{outer} = F \ in \ \Omega_{outer}, \\ \phi^p_{outer}(0) = \alpha_0 \ , \ \phi^p_{outer}(a) = \phi^p_{inner}(a), \\ L^{h_2} \phi^{p+1}_{inner} = F \ in \ \Omega_{inner}, \\ \phi^{p+1}_{inner}(0) = \alpha_1 \ , \\ \dfrac{\phi^{p+1}_{inner}(a+h_2) - \phi^{p+1}_{inner}(a)}{h_2} = \dfrac{\phi^p_{outer}(a) - \phi^p_{outer}(a-h_1)}{h_1}. \end{cases} \tag{3.3}$$

To start the scheme, we impose an artificial boundary condition at the point a. We use the same finite difference scheme in each subdomain with Dirichlet boundary condition at a in Ω_{outer} and Neumann boundary condition at a in Ω_{inner}. We will consider in the next

section the other possible choice, i.e Neumann boundary condition at a in Ω_{outer} and the Dirichlet boundary condition at a in Ω_{inner}.

We will decompose the analysis of this iterative method in *three steps*: first we define the best interface location between the subdomains, based on a truncation error analysis, secondly we derive from the stability property of the discretized operator the rate of damping of the artificial boundary condition error. Lastly we combine these two results to get an estimate of convergence of the iterative solver to the *exact* solution of the differential problem (3.1).

The technique of demonstration is quit elementary but plays with two types of small parameters: first the space steps (3.2), second the small singular perturbation parameter ϵ. Our goal is to find the best path in the parameter space (h,ϵ) which provides superlinear convergence and optimal uniform approximation.

– First Step : interface position

We wish to determine the optimal interface position a, which minimizes the maximum error in both subdomains under the constraint that we have the same asymptotic order of mesh points N inside each subdomain. In this part, we neglect the artificial boundary condition error inherent to the Funaro-Quarteroni alternate $(F.Q)$ procedure. This error will be taken care of later on.

Let ϕ_{outer}(respt ϕ_{inner}) be the restriction of ϕ to Ω_{outer} (respt Ω_{inner}). We define the following errors:

$$e_{outer} = \max_{\Omega_{outer}} |\phi_{outer} - \phi_{outer}^{h_1}|$$
$$e_{inner} = \max_{\Omega_{inner}} |\phi_{inner} - \phi_{inner}^{h_2}|.$$

A centered-order finite difference scheme applied to $-\epsilon u'' + u = f$ with exact Dirichlet boundary conditions gives

$$e_{outer} \approx \epsilon h_1^2 a^2 \max_{\Omega_{outer}} |\frac{d^{(4)}\phi}{dx^4}|. \tag{3.4}$$

The analysis of the inner subdomain approximation with mixed exact boundary conditions gives two truncation errors, which we should consider in addition the discretisation error of the Neumann boundary condition. We have

$$e_{inner} \approx \epsilon h_2^2 (1-a)^2 \max_{\Omega_{inner}} |\frac{d^{(4)}\phi}{dx^4}| + h_2^2 \frac{R}{1-R} \max_{\Omega_{inner}} |\frac{d^{(2)}\phi}{dx^2}|, \tag{3.5}$$

where $R = 1 + \frac{h_2^2}{2\epsilon} + \frac{h_2}{\sqrt{\epsilon}}(1 + \frac{h_2^2}{2\epsilon})^{\frac{1}{2}}$.

We first notice that the truncation errors defined above depend strongly on the property of the solution that we want to approximate in each subdomain.

Let ϕ_0 be the outer expansion of ϕ and $\Theta(x,\epsilon)$ be the corrector i.e

$$\Theta(x,\epsilon) = \phi(x,\epsilon) - \phi_0(x,\epsilon) \approx exp(-\eta),$$

in the boundary layer with $\eta = \frac{1-x}{\epsilon}$. We will show that the truncation error is dominated by the behavior of the corrector as in ([28]).

Secondly we remark that the error in both subdomains is coupled because the Neumann boundary condition for the *inner* domain is only an approximation of a derivative in the *outer* domain. So we need to compute directly the error between the exact solution of the continuous problem and the formal limit of (3.3) when $p \to \infty$. We have:

Lemme 2 *Let $\tilde{\phi} = (\tilde{\phi}_{i,j})_{i=0,\dots N, j=1,2}$ be the solution of the following linear system:*

$$\begin{cases} L^{h_1}\tilde{\phi}_{i,1} = F \quad i = 1, \dots N-1, \\ L^{h_2}\tilde{\phi}_{i,2} = F \quad i = 1, \dots N-1, \\ \tilde{\phi}_{0,1} = \alpha_0 \ , \ \tilde{\phi}_{N,1} = \tilde{\phi}_{0,2}, \ \tilde{\phi}_{N,2} = \alpha_1, \\ \dfrac{\tilde{\phi}_{1,2} - \tilde{\phi}_{0,2}}{h_2} = \dfrac{\tilde{\phi}_{N,1} - \tilde{\phi}_{N-1,1}}{h_1}, \end{cases} \tag{3.6}$$

where

$$L^h\phi_i = -\epsilon\frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{h} + \phi_i.$$

Let M be the composit grid $M = M_{outer} \cup M_{inner}$, with

$$\begin{cases} M_{outer} = \{x_{i,1} = i(\frac{a}{N}); i = 0, \dots N\} \\ M_{inner} = \{x_{i,2} = a + i(\frac{1-a}{N}); i = 0, \dots N\}. \end{cases}$$

Let us suppose that

$$N^{-1} \approx \sqrt{\epsilon}\delta \qquad with \qquad \delta \succ\succ 1.$$

Let $\| \ \|_\infty$ be the maximum norm on the composite grid M.

Under the previous hypothesis on the discretization and approximation of the operators in each subdomain, $\| \ \phi - \tilde{\phi}\|_\infty$ is asymptotically minimum when

$$1 - a \sim \sqrt{\epsilon}\log(\epsilon^{-2}).$$

Preuve: The existence and uniqueness of $\tilde{\phi}$ follows from the maximum principle. Let ϕ be the exact solution of the Dirichlet problem: $-\epsilon\phi'' + \phi = F$; $\phi(0) = \alpha_0$; $\phi(1) = \alpha_1$.

Let $e_{i,j} = \phi_{i,j} - \tilde{\phi}_{i,j} \quad i = 0, \dots N, j = 1,2$, where $\phi_{i,j}$ is the trace of ϕ on the composit grid $M_{outer} \cup M_{inner}$.
We have

$$\begin{cases} -\epsilon\dfrac{e_{i+1,1} - 2e_{i,1} + e_{i-1,1}}{h_1^2} + e_{i,1} = -\epsilon\dfrac{h_1^2}{12}\phi^{(4)}(\xi_i), \\ -\epsilon\dfrac{e_{i+1,2} - 2e_{i,2} + e_{i-1,2}}{h_2^2} + e_{i,2} = -\epsilon\dfrac{h_2^2}{12}\phi^{(4)}(\eta_i), \\ e_{0,1} = 0 \ , \ e_{N,2} = 0, \\ e_{N,1} = e_{0,2}, \\ \dfrac{e_{1,2} - e_{0,2}}{h_2} = \dfrac{e_{N,1} - e_{N-1,1}}{h_1} + \dfrac{h_1}{2}\phi^{(2)}(\tilde{\xi}) + \dfrac{h_2}{2}\phi^{(2)}(\tilde{\eta}), \end{cases}$$

where

$$x_{i-1,1} < \xi_i < x_{i,1}; \ x_{i-1,2} < \eta_i < x_{i,2}; \ x_{N-1,1} < \tilde{\xi} < x_{N,1}; \ x_{0,2} < \tilde{\eta} < x_{1,2}.$$

We split the error $e_{i,j}$ into two componants $\tilde{e}_{i,j}$ and $\hat{e}_{i,j}$ solutions of the following linear systems :

$$\begin{cases} -\epsilon\dfrac{\tilde{e}_{i+1,1}-2\tilde{e}_{i,1}+\tilde{e}_{i-1,1}}{h_1^2}+\tilde{e}_{i,1}=-\epsilon\dfrac{h_1^2}{12}\phi^{(4)}(\xi_i), \\ -\epsilon\dfrac{\tilde{e}_{i+1,2}-2\tilde{e}_{i,2}+\tilde{e}_{i-1,2}}{h_2^2}+\tilde{e}_{i,2}=-\epsilon\dfrac{h_2^2}{12}\phi^{(4)}(\eta_i), \\ \tilde{e}_{0,1}=0\ ,\ \tilde{e}_{N,2}=0, \\ \tilde{e}_{N,1}=\tilde{e}_{0,2}, \\ \dfrac{\tilde{e}_{1,2}-\tilde{e}_{0,2}}{h_2}=\dfrac{\tilde{e}_{N,1}-\tilde{e}_{N-1,1}}{h_1}, \end{cases}$$

$$\begin{cases} -\epsilon\dfrac{\hat{e}_{i+1,1}-2\hat{e}_{i,1}+\hat{e}_{i-1,1}}{h_1^2}+\hat{e}_{i,1}=0, \\ -\epsilon\dfrac{\hat{e}_{i+1,2}-2\hat{e}_{i,2}+\hat{e}_{i-1,2}}{h_2^2}+\hat{e}_{i,2}=0, \\ \hat{e}_{0,1}=0\ ,\ \hat{e}_{N,2}=0, \\ \hat{e}_{N,1}=\hat{e}_{0,2}, \\ \dfrac{\hat{e}_{1,2}-\hat{e}_{0,2}}{h_2}=\dfrac{\hat{e}_{N,1}-\hat{e}_{N-1,1}}{h_1}+\dfrac{h_1}{2}\phi^{(2)}(\tilde{\xi})+\dfrac{h_2}{2}\phi^{(2)}(\tilde{\eta}). \end{cases}$$

If $|\tilde{e}_{i,1}|$ is not maximal at the boundary $i=N$ then

$$\max_{i=0,\dots N}|\tilde{e}_{i,1}|\leq E_1=C\epsilon h_1^2\max\{\phi^{(4)}\}.$$

Otherwise, we deduce from

$$e_{1,2}-e_{0,2}=\frac{h_2}{h_1}(e_{N,1}-e_{N-1,1}),$$

that $|e_{i,2}|$ cannot be maximal at the boundary $i=0$. We have therefore

$$\max|e_{i,2}|\leq E_2=C(1-a)^2\epsilon h_2^2\max\{\phi^{(4)}\}.$$

Consequently, we have :

$$\|\tilde{e}\|_\infty\leq\max(E_1,E_2).$$

Now let us consider the error that comes from the discretization of the Neumann boundary condition.
From the maximum principle, we have $|\hat{e}_{i,1}|$ bounded by $|\hat{e}_{N,1}|=|\hat{e}_{0,2}|$. In addition ([28]-lemme 4) :

$$\frac{\hat{e}_{N,1}-\hat{e}_{N-1,1}}{h_1}\approx\frac{\hat{e}_{0,2}}{h_1}.$$

We look for $\hat{e}_{i,2}$ in the following form :

$$\hat{e}_{i,2}=C_1R^i+C_2R^{-i},$$

where $R(h_2,\epsilon)$ is the larger root of the quadratic polynomial

$$R^2-(2+\frac{h_2^2}{\epsilon})R+1=0.$$

We obtain C_1 and C_2 from the boundary conditions. From this explicit formula we can conclude that $\hat{e}_{i,2}$ is a decreasing function of i and that:

$$\hat{e}_{0,2} \approx E_3 = \sqrt{\epsilon}\frac{h_1}{2}\phi^{(2)}(b).$$

We have finally:

$$\|e\|_\infty \leq \max(E_1,E_2,E_3), \quad with \tag{3.7}$$

$$E_1 \approx \epsilon h_1^2(1+\epsilon^{-2}\exp(-\frac{b}{\sqrt{\epsilon}}), \tag{3.8}$$

$$E_2 \approx \epsilon^{-1}h_1^2 b^4, \tag{3.9}$$

$$E_3 \approx \sqrt{\epsilon}h_1(1+\epsilon^{-1}\exp(-\frac{b}{\sqrt{\epsilon}}). \tag{3.10}$$

Hence, if ϵ is sufficiently small then E_2 is an increasing function of b and E_1 as well as E_3 are decreasing functions of b. It is easy to show that the accuracy is best when

$$b \sim \sqrt{\epsilon}\log\epsilon^{-2}.$$

The error is then

$$\|e\|_\infty \approx \epsilon\delta. \tag{3.11}$$

■

– Second Step: Damping of artificial boundary errors

The convergence of the method depends essentially on the way an error which is introduced at the artificial interface propagates inside the subdomain.

The Schwarz iterate procedure applied to the two points boundary value problem $-u'' = F$, $u(0) = \alpha_0$, $u(1) = \alpha_1$, for example, has very poor efficiency because for $[a,b] \subset [0,1]$ any given subdomain of [0,1] used in the Schwarz iterative procedure, any perturbation of the boundary condition at b decreases linearly inside the subdomain [a,b]. It is no longer the case for a second order singular perturbation problem that have boundary layers with fast exponentially decay. In [28] it is shown that we may, in fact, have fast convergence with relatively small overlap even if we apply the straightforward Schwarz alternate procedure with Dirichlet boundary conditions.
We will proove that the F.Q procedure may also have fast convergence and we will compare it with the Schwarz alternating procedure.

Let us consider the F.Q iterative procedure applied to the following homogeneous problem:

$$\begin{cases} L_1^h e_{i,1}^p = 0\ , \\ e_{0,1}^p = 0\ ,\ e_{N,1}^p = e_{0,2}^{p-1}\ , \\ L_2^h e_{N,2}^p = 0\ , \\ \dfrac{e_{1,2}^p - e_{0,2}^p}{h_2} = \dfrac{e_{N,1}^p - e_{N-1,1}^p}{h_1}\ , \\ e_{N,2}^p = 0\ , \end{cases}$$

with a domain decomposition given by (Lemma 2), i.e $b = 1 - a \approx \sqrt{\epsilon}\log\epsilon^{-1}$. The discretized operator satisfies a maximum principle and we can show that:

$$|e^p_{0,2}| = |e^{p+1}_{N,1}| \geq max_{i=0,\dots N-1}|e^{p+1}_{i,1}|$$

and

$$|e^p_{0,2}| \geq max_{i=0,\dots N}|e^p_{i,2}|.$$

We will call damping factor a real ξ such that: $|e^{p+1}_{0,2}| \leq \xi|e^p_{0,2}|$, $\forall p$.

Lemme 3 *Let a be the interface position between the subdomains such that $1 - a \approx \epsilon^{1/2}\log\epsilon^{-1}$. Suppose that $N^{-1} \approx \epsilon^{\frac{1}{2}}\delta,$. with $\delta \succ\succ 1$.*
Then the amplification factor of the iterative scheme is:

$$\xi \approx \delta^{-1}.$$

Preuve: Let $\tilde{\phi}^p = (\tilde{\phi}^p_{i,j})_{i=0,\dots N,j=1,2}$ be the solution of

$$\begin{cases} L^{h_1}\tilde{\phi}^p_{i,1} = F \quad i = 1,\dots N-1, \\ L^{h_2}\tilde{\phi}^p_{i,2} = F \quad i = 1,\dots N-1, \\ \tilde{\phi}^p_{0,1} = \alpha_0 \ ; \ \tilde{\phi}^p_{N,1} = \tilde{\phi}^{p-1}_{0,2} \ ; \ \tilde{\phi}^p_{N,2} = \alpha_1, \\ \dfrac{\tilde{\phi}^p_{1,2} - \tilde{\phi}^p_{0,2}}{h_2} = \dfrac{\tilde{\phi}^p_{N,1} - \tilde{\phi}^p_{N-1,1}}{h_1}. \end{cases}$$

Let $(V^p_{i,j})_{i=0,\dots N,j=1,2}$ defined as $V^p_{i,j} = \tilde{\phi}^p_{i,j} - \tilde{\phi}_{i,j}$ where $\tilde{\phi} = (\tilde{\phi}_{i,j})_{i=0,\dots N,j=1,2}$ is the solution of (3.6).

We have:

$$\begin{cases} L^{h_1}V^p_{i,1} = 0 \quad i = 1,\dots N-1, \\ L^{h_2}V^p_{i,2} = 0 \quad i = 1,\dots N-1, \\ V^p_{0,1} = 0 \ , \ V^p_{N,1} = V^{p-1}_{0,2} \ , \ V^p_{N,2} = 0, \\ \dfrac{V^p_{1,2} - V^p_{0,2}}{h_2} = \dfrac{V^p_{N,1} - V^p_{N-1,1}}{h_1}. \end{cases}$$

Let $\xi_{\leftarrow}$ be the damping factor for the outer domain with Dirichlet boundary condition defined as in [28], we have $V^p_{N-1,1} = \xi_{\leftarrow}V^p_{N,1}$ with $\xi_{\leftarrow} \prec\prec 1$. Then

$$V^p_{1,2} - V^p_{0,2} = \frac{h_2}{h_1}(V^p_{N,1} - V^p_{N-1,1}) = \frac{h_2}{h_1}(1 - \xi_{\leftarrow})V^p_{N,1} = \frac{h_2}{h_1}(1 - \xi_{\leftarrow})V^{p-1}_{0,2}. \tag{3.12}$$

Let K be the RHS of (3.12), we will show that $(V^p_{0,2})_p$ is decreasing in module. $V^p_{i,2}$ can be writed as

$$V^p_{i,2} = C^p_1 R^i(h_2,\epsilon) + C^p_2 R^{-i}(h_2,\epsilon),$$

where $R(h_2,\epsilon)$ is the larger root of the quadratic polynomial

$$R^2 - (2 + \frac{h_2^2}{\epsilon})R + 1 = 0.$$

From the boundary conditions, we obtain C_1^p and C_2^p

$$\begin{cases} C_2^p = -C_1^p R^{2N}(h_2,\epsilon) \\ C_1^p = K/(R(h_2,\epsilon)-1)(1+R^{2N-1}(h_2,\epsilon)). \end{cases}$$

We can write

$$\begin{aligned} V_{0,2}^p &= \frac{1-R^{2N}(h_2,\epsilon)}{1+R^{2N}(h_2,\epsilon)} \times \frac{K}{R(h_2,\epsilon)-1} \\ &= \xi \times V_{0,2}^{p-1}, \end{aligned}$$

where ξ is *the amplificator factor of the iterative method.* Consequently

$$\xi = \frac{h_2}{h_1}(1-\xi_{\leftarrow})\frac{1-R^{2N}(h_2,\epsilon)}{1+R^{2N-1}(h_2,\epsilon)} \times \frac{1}{R(h_2,\epsilon)-1}.$$

From $N \approx \epsilon^{\frac{1}{2}}\delta$, we deduce the asymptotic behavior of ξ which is

$$\xi \approx \delta^{-1}.$$

■

- **third step : Convergence to the solution of the ODE problem and Uniform approximation**

Théorème 3 *Let ϕ be the solution of the Dirichlet problem*

$$L[\phi] = -\epsilon\phi'' + \phi = F \ , \ \phi(0) = \alpha_0, \ \phi(1) = \alpha_1.$$

Let ϕ_{outer}^p and ϕ_{inner}^p defined by the iterative scheme:

$$\begin{cases} L^{h_1}\phi_{outer}^p = F \text{ in } \Omega_{outer}, \\ \phi_{outer}^p(0) = \alpha_0 \ , \ \phi_{outer}^p(a) = \phi_{inner}^p(a), \\ L^{h_2}\phi_{inner}^{p+1} = F \text{ in } \Omega_{inner}, \\ \phi_{inner}^{p+1}(0) = \alpha_1, \\ \dfrac{\phi_{inner}^{p+1}(a+h_2) - \phi_{inner}^{p+1}(a)}{h_2} = \dfrac{\phi_{outer}^p(a) - \phi_{outer}^p(a-h_1)}{h_1}, \end{cases}$$

where

$$L^h[\phi] = -\epsilon\frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{h^2} + \phi_i.$$

Let

$$\phi^p = \begin{cases} \phi_{outer}^p & on \quad M_{outer} = \{x_{i,1} = i(\frac{a}{N}); i = 0, \dots N\} \\ \phi_{inner}^p & on \quad M_{inner} = \{x_{i,2} = a + i(\frac{1-a}{N}); i = 0, \dots N\}. \end{cases}$$

Let $\| \ \|_\infty$ be the maximum norm on the composite grid $M_{outer} \cup M_{inner}$.
Let us suppose that

$$N^{-1} \approx \sqrt{\epsilon}\delta \quad with \quad \delta \succ\succ 1 \qquad and \qquad b \sim \sqrt{\epsilon}\log(\epsilon^{-2}).$$

Then

$$\|\phi - \phi^p\|_\infty \leq C(\xi^p + \epsilon\delta)), \ \xi \approx \delta^{-1}. \tag{3.13}$$

Preuve: We have shown in Lemma 2 that $e_{i,j} = \phi_{i,j} - \tilde{\phi}_{i,j}$ $i = 0, \ldots N, j = 1,2$, where ϕ is the exact solution of the Dirichlet problem (3.1), $\phi_{i,j}$ is the trace of ϕ on the composit grid $M_{outer} \cup M_{inner}$ and $\tilde{\phi} = (\tilde{\phi}_{i,j})_{i=0,\ldots N, j=1,2}$ solution of (3.6) satisfies (3.11).

Let $\tilde{\phi}_{outer} = (\tilde{\phi}_{i,1})_{i=0,\ldots N}$ (respt $\tilde{\phi}_{inner} = (\tilde{\phi}_{i,2})_{i=0,\ldots N}$) and $V^p_{outer} = \phi^p_{outer} - \tilde{\phi}_{outer}$ (respt $V^p_{inner} = \phi^p_{inner} - \tilde{\phi}_{inner}$).
We have:

$$\begin{cases} L^{h_1}(V^p_{outer}) = 0 \quad \text{on } \Omega_{outer}, \\ V^p_{outer}(a) = V^{p-1}_{inner}(a) \quad , \quad V^p_{outer}(0) = 0, \\ L^{h_2}(V^p_{inner}) = 0 \quad \text{on } \Omega_{inner}, \\ V^p_{inner}(1) = 0, \\ \dfrac{V^p_{inner}(a+h_2) - V^p_{inner}(a)}{h_2} = \dfrac{V^p_{outer}(a) - V^p_{outer}(a-h_1)}{h_1}. \end{cases}$$

We conclude from lemma 3 that:

$$\|V^p_{outer}\|_\infty \leq C\,\xi^p \text{ on } \mathrm{M_{outer}},\ \|\mathrm{V^p_{inner}}\|_\infty \leq \mathrm{C}\,\xi^{\mathrm{p}} \text{ on } \mathrm{M_{inner}},$$

and therefore

$$\|\phi - \phi^p\|_\infty \leq C(\xi^p + \epsilon\delta).$$

■

According to the estimate of (3), we observe that p must be choosen such that

$$\xi^p \approx \max(e_{outer}, e_{inner}).$$

It is interesting to notice that the convergence speed of the F.Q procedure is not as good as the convergence speed of the straightforward Schwarz alternate procedure with minimum overlap. For $N^{-1} \approx \sqrt{\epsilon}\log(\epsilon^{-1})$, we have $\xi \approx \epsilon log^{-2}(\epsilon^{-1})$ ([28], Theorem 1), instead of $\xi \approx log^{-1}(\epsilon^{-1})$. But on the other side the F.Q iterative procedure is a *non-overlapping* domain decomposition method.

Let us apply the present Dirichlet-Neumann scheme to the simple model problem

$$-\epsilon u'' + u = (1+\epsilon)\cos(x), x \in (0,1),\ u(0) = 1,\ u(1) = 1 + cos(1). \tag{3.14}$$

Figure 1 shows the dependence of the error in maximum norm as a function of the interface position with two times 20 discretization points. We have checked that the optimal interface position corresponds to the balance of the error in both subdomains. Figure 2 demonstrates that the smaller is ϵ, the faster is the convergence of the iterative procedure toward the solution of the discretized problem. We have completed our numerical investigation with the original F.Q. scheme which includes a relaxation on the interface condition [26]; Figure 3 shows the effect of this relaxation and we observe that the optimal relaxation factor goes to one as $\epsilon \to 0$.

Neumann-Dirichlet scheme

We are going to show that the choice of the boundary conditions at the artificial interface a is critical. Let us consider now the F.Q method with Neumann boundary

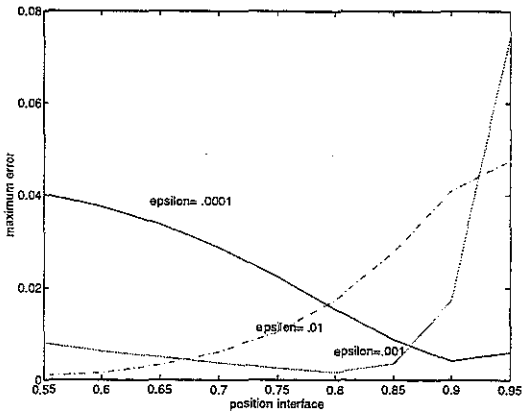


FIG. 3.1 – *Effect of position interface on maximum error*

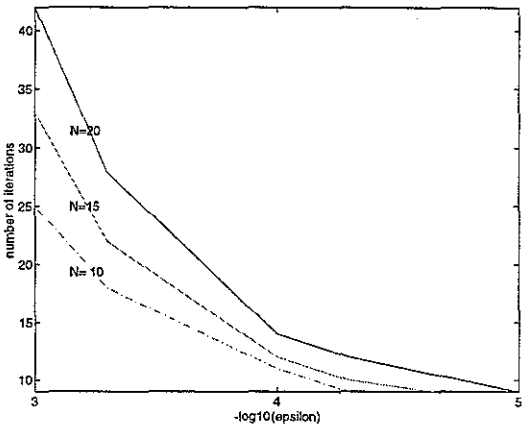


FIG. 3.2 – *number of iterations (with no relaxation)*

condition at a in Ω_{outer} and the Dirichlet boundary condition at a in Ω_{inner}. The scheme gives :

$$\begin{cases} L^{h_1}\phi^p_{outer} = F \text{ on } \Omega_{\text{outer}}, \\ \dfrac{\phi^{p+1}_{outer}(a) - \phi^{p+1}_{outer}(a-h_1)}{h_1} = \dfrac{\phi^p_{inner}(a+h_2) - \phi^p_{inner}(a)}{h_2}\ ,\ \phi^p_{outer}(0) = \alpha_0, \\ L^{h_2}\phi^p_{inner} = F \text{ on } \Omega_{\text{inner}}, \\ \phi^p_{inner}(1) = \alpha_1\ ,\ \phi^p_{inner}(a) = \phi^p_{outer}(a). \end{cases} \tag{3.15}$$

To start the scheme, we impose an artificial boundary condition at point a. The best choice for the interface position in terms of accuracy is $1-a \approx \sqrt{\epsilon}\log\epsilon^{-1}$ since the formal limit of (3.15) is identique to the formal limit of (3.3). But, we are going to show that this procedure is then highly unstable.

Théorème 4 *Let us assume that* $h_1 \succ\succ h_2$ *and* $h_1 \succ\succ \sqrt{\epsilon}$.
Then the amplificator factor of the iterative procedure satisfies

$$\xi \sim \frac{h_1}{h_2},$$

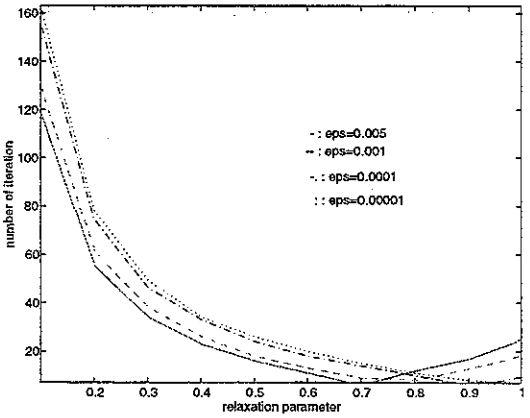


FIG. 3.3 – *effect of relaxation parameter on number of iterations*

and the F.Q procedure with no relaxation is highly unstable.

Preuve: We only need to consider the F.Q procedure applied to the problem with homogeneous boundary condition :

$$\begin{cases} L^{h_1} V^p_{i,1} = 0 \quad i = 1, \dots N-1, \\ L^{h_2} V^p_{i,2} = 0 \quad i = 1, \dots N-1, \\ V^p_{0,1} = 0 \ , \ V^p_{0,2} = V^p_{N,1} \ , \ V^p_{N,2} = 0, \\ \dfrac{V^p_{N,1} - V^p_{N-1,1}}{h_1} = \dfrac{V^{p-1}_{1,2} - V^{p-1}_{0,2}}{h_2}. \end{cases}$$

We have

$$V^p_{1,2} = \xi_{\leftarrow} V^p_{0,2}$$

with $\xi_{\leftarrow} \prec\prec 1$. Then

$$V^p_{N,1} - V^p_{N-1,1} = \frac{h_1}{h_2}(V^p_{1,2} - V^p_{0,2}) = \frac{h_1}{h_2}(1 - \xi_{\leftarrow}) V^{p-1}_{0,2} = \frac{h_1}{h_2}(1 - \xi_{\leftarrow}) V^{p-1}_{N,1}. \tag{3.16}$$

Let K be the RHS of (3.16), we will show that $(V^p_{N,1})_p$ is divergente. $V^p_{i,1}$ can be writed as

$$V^p_{i,1} = C^p_1 R^i(h_1,\epsilon) + C^p_2 R^{-i}(h_1,\epsilon),$$

where $R(h_1,\epsilon)$ is the larger root of the quadratic polynomial

$$R^2 - (2 + \frac{h_1^2}{\epsilon})R + 1 = 0.$$

From the boundaries conditions we determine C^p_1 and C^p_2; we have

$$V^p_{N,1} = \frac{1 - R^{2N}(h_1,\epsilon)}{1 + R^{2N}(h_1,\epsilon)} \times \frac{K}{R(h_1,\epsilon) - 1} V^p_{0,2} = \xi V^{p-1}_{N,1}.$$

ξ *the amplificator factor* of the iterative methode is then

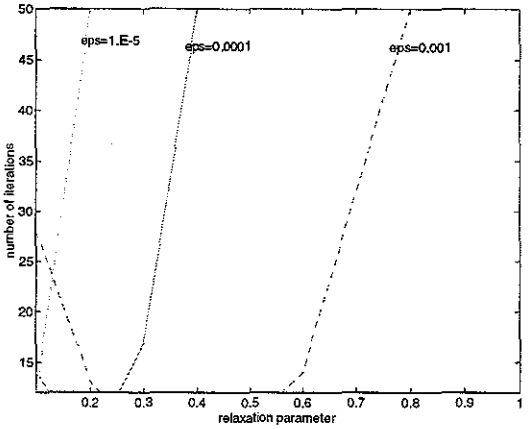
$$\xi = \frac{h_1}{h_2}(1 - \xi_{\leftarrow}) \frac{1 - R^{2N}(h_1,\epsilon)}{1 + R^{2N-1}(h_1,\epsilon)} \times \frac{1}{R(h_1,\epsilon) - 1} V^{p-1}_{N,1}.$$

We conclude easily

$$\xi \sim \frac{h_1}{h_2} \succ\succ 1.$$

■

It is quit possible to introduce a relaxation on the artificial boundary condition as in [26] to retrieve the convergence of F.Q method, however the value of this parameter is not easy to find in general. Our experiment with (3.14 : cf Figure 4) shows that this relaxation factor goes to zero as $\epsilon \to 0$.



UNIVERSITÉ NANCY 1 BIBLIOTHÈQUE SCIENCES

FIG. 3.4 – *effect of relaxation parameter on convergence*

3.2.2 Heterogeneous domain decomposition

In this section, we consider a linear second-order transmission problem of the following type:

$$\begin{cases} L_1\phi = -\epsilon\phi'' + \phi = F \text{ in } \Omega_1 = (0,A), \\ L_2\psi = \psi'' = G \text{ in } \Omega_2 = (A,1), \\ \phi(A) = \psi(A),\ \phi'(A) = \psi'(A), \\ \phi'(0) = 0\ ,\ \psi(1) = 0. \end{cases} \tag{3.17}$$

Where ϵ is a small positive parameter, $\epsilon \in]0,\epsilon_0]$ for some $\epsilon_0 > 0$. In addition we assume the compatibility condition : all derivatives of F vanish in 0. This very simple model is introduced to study the convergence of an heterogeneous domain decomposition based on the F.Q method. We observe that the domain decomposition is dictated by the definition of the transmission problem and that there is no overlap of the subdomains on A.

Asymptotic analysis

Let us study the boundary layer of (3.17). We define the regular asmptotic expansion of ϕ in Ω_1 as

$$\phi_R = \phi_0 + \sqrt{\epsilon}\phi_1 + \epsilon\phi_2 + \epsilon^{\frac{3}{2}}\phi_3 + ...$$

This regular expansion is completely determined from the PDE equation

$$L_1\phi_R = -\epsilon\phi_R^{"} + \phi_R = F \ in \ \Omega_1 = (0,A);$$

we have formally $\phi_0 = F$, $\phi_1 = F^{(2)}$, ... $\phi_i = F^{(2i)}$,... ϕ_R satisfies the homogeneous Neumann boundary condition at 0, but not necessarily the transmission condition at A. We therefore introduce a corrector ϕ_C depending on the local stretched variable $\eta = \frac{x-A}{\sqrt{\epsilon}}$.

At first order we have

$$\phi_0(A) + \phi_C(A) = \psi(A); \ \phi_0^{'}(A) + \epsilon^{-\frac{1}{2}}\phi_C^{'} = \psi^{'}(A).$$

These transmission conditions suggest that the corrector and the solution in domain Ω_2 have the following expansion structures

$$\psi = \psi_0 + \sqrt{\epsilon}\psi_1 + \epsilon\psi_2 + \epsilon^{\frac{3}{2}}\psi_3 + ,\ldots \ and \ \phi_C = \sqrt{\epsilon}\psi_C^0 + \epsilon\psi_C^1 + \epsilon^{\frac{3}{2}}\psi_C^2 + \ldots$$

It is then easy to build a uniform formal asymptotic expansion at arbitrary high order with the following chain rule

$$\begin{cases} L_2\psi_i = \delta_i^0 G \ in \ \Omega_2, \\ \psi_i(A) = \phi_i(A) - \delta_i^0\phi_C^{i-1}(0), \\ \psi_i(1) = 0, \end{cases} \tag{3.18}$$

and

$$\begin{cases} -(\phi_C^i)^{"} + \phi_C^i = 0 \ in \ \Omega_1, \\ \phi_C^i(\eta) \to 0 \ when \ \eta \to -\infty, \\ (\phi_C^i)^{'}(0) = \psi_i'(A) - \phi_i'(A), \end{cases} \tag{3.19}$$

with $\delta_i^0 = 0$(respt1) if $i = 0$(respt $i > 0$) and 1 if $i > 0$. Since each term of these expansions and its derivatives is bounded independantly of ϵ, we conclude that the formal asymptotic expansion is consistant with (3.17) at arbitrary high order. From the energy estimate

$$\int_0^A \epsilon\phi^{'2} + \phi^2 dx + \int_A^1 \epsilon\psi^{'2} dx = \int_0^A F\phi dx - \int_A^1 \epsilon G dx,$$

we obtain the stability bound

$$\max\left(\max_{(0,A)}\phi, \max_{(A,1)}\psi\right) \leq \max\left(\epsilon^{-1}||F||_\infty, ||G||_\infty\right), \tag{3.20}$$

and conclude the validity of the formal asymptotic expansion. (3.17) is consequently a singular perturbation problem with a weak layer of $\sqrt{\epsilon}$ thikness located to the left of A.

Numerical procedure

The asymptotic analysis suggests that the domain computation should be split into three subdomains $\Omega_1 = (O,B)$, $\Omega_2 = (B,A)$ and $\Omega_3 = (A,1)$ where the intermediate subdomain is used to resolve the boundary layer. We assume that F vanishes in the

neighbourhod of zero and that the space step h_i for each subdomain satisfies the following asymptotic relation :

$$\frac{h_1}{B} \approx \frac{h_2}{A-B} \approx h_3 \approx \frac{1}{N}.$$

with $b = A - B \prec\prec 1$. We are going to study the heterogeneous F.Q procedure for such problem.

First scheme :

We choose first the F.Q procedure to resolve the layer and the transmission problem and according to the previous results, we adopt the Dirichlet - Neumann scheme for the layer and the Neumann - Dirichlet scheme for the transmission problem. The iterative procedure gives :

$$\left\{\begin{array}{l} L_1^{h_1}\phi_1^p = F \ in \ \Omega_1, \\ \phi_1^p(0) = \phi_1^p(h_1) \ , \ \phi_1^p(B) = \phi_2^p(B), \\ L_2^{h_3}\psi^p = G \ in \ \Omega_3, \\ \psi_3^p(A) = \phi_2^p(A) \ , \ \psi_3^p(1) = 0, \\ L_1^{h_2}\phi_2^{p+1} = F \ in \ \Omega_2, \\ \dfrac{\phi_2^{p+1}(B+h_2) - \phi_2^{p+1}(B)}{h_2} = \dfrac{\phi_1^p(B) - \phi_1^p(B-h_1)}{h_1}, \\ \dfrac{\phi_2^{p+1}(A) - \phi_2^{p+1}(A-h_2)}{h_2} = \dfrac{\psi^p(A+h_3) - \psi^p(A)}{h_3}. \end{array}\right. \tag{3.21}$$

The proof of convergence of this scheme is very similar to Sect 2.1.1. We obtain first the optimal interface position with :

Lemme 4 *Let $(\tilde{\phi},\tilde{\psi})$ with $\tilde{\phi} = (\tilde{\phi}_{i,j})_{i=0,\dots N, j=1,2}$ and $\tilde{\psi} = (\tilde{\psi}_i)_{i=0,\dots N}$ be the solution of the linear system that is the formal limit of (3.21) when $p \to \infty$.*

Let M be the composite grid $M = M_1 \cup M_2 \cup M_3$, with

$$\left\{\begin{array}{l} M_1 = \{x_{i,1} = i(\frac{B}{N}); i = 0, \dots N\} \\ M_2 = \{x_{i,2} = B + i(\frac{A-B}{N}); i = 0, \dots N\} \\ M_3 = \{x_{i,2} = A + i(\frac{1-A}{N}); i = 0, \dots N\}. \end{array}\right.$$

Let us suppose that $N^{-1} \approx \sqrt{\epsilon}\delta$, with $\delta \succ\succ 1$. Let $\| \ \|_\infty$ be the maximum norm on the composite grid M.

Under the previous hypothesis on the discretization and approximation of the operators in each subdomain, $\max(\| \phi - \tilde{\phi} \|_\infty, \| \psi - \tilde{\psi} \|_\infty)$ *is asymptotically minimum when*

$$b = A - B \approx \sqrt{\epsilon}\log(\epsilon^{-1}).$$

Preuve: The existence and uniqueness of $\tilde{\phi}$ follows from the maximum principle. Let (ϕ,ψ) be the exact solution of the Dirichlet problem (3.17). Let $e_{i,j} = \phi_{i,j} - \tilde{\phi}_{i,j} \quad i = 0, \dots N, j = 1, \dots 2, e_{i,3} = \psi_i - \tilde{\psi}_i \quad i = 0, \dots N$, where $\phi_{i,j}$ (respt ψ_i) is the trace of ϕ (respt

ψ) on the composit grid M. We have

$$-\epsilon\frac{e_{i+1,1}-2e_{i,1}+e_{i-1,1}}{h_1^2}+e_{i,1} = -\epsilon\frac{h_1^2}{12}\phi^{(4)}(\xi_i), \quad (3.22)$$

$$-\epsilon\frac{e_{i+1,2}-2e_{i,2}+e_{i-1,2}}{h_2^2}+e_{i,2} = -\epsilon\frac{h_2^2}{12}\phi^{(4)}(\eta_i), \quad (3.23)$$

$$\frac{e_{i+1,3}-2e_{i,2}+e_{i-1,3}}{h_2^3} = \frac{h_2^3}{12}\psi^{(4)}(\zeta_i), \quad (3.24)$$

$$e_{1,1}=e_{0,1}\ ,\ e_{N,3}=0,$$

$$e_{N,1}=e_{0,2}\ ,\ e_{N,2}=e_{0,3},$$

$$\frac{e_{1,2}-e_{0,2}}{h_2}-\frac{e_{N,1}-e_{N-1,1}}{h_1} = \frac{h_1}{2}\phi^{(2)}(\tilde{\xi})+\frac{h_2}{2}\phi^{(2)}(\tilde{\eta}), \quad (3.25)$$

$$\frac{e_{1,3}-e_{0,3}}{h_3}-\frac{e_{N,2}-e_{N-1,2}}{h_2} = \frac{h_2}{2}\phi^{(2)}(\hat{\xi})+\frac{h_3}{2}\psi^{(2)}(\hat{\eta}), \quad (3.26)$$

where

$$x_{i-1,1}<\xi_i<x_{i,1};\ x_{i-1,2}<\eta_i<x_{i,2};\ \ldots$$

$$x_{N-1,1}<\tilde{\xi}<x_{N,1};\ x_{0,2}<\tilde{\eta}<x_{1,2};\ldots$$

Using the superposition principle, we can compute independently the error contribution of each RHS of (3.22), (3.23), (3.24), (3.25) and (3.26). We have

$$\|e\|_\infty \leq \max_{i=1,\ldots 5}(E_i),$$

with

$$E_1 \approx \epsilon h_1^2(1+\epsilon^{-\frac{3}{2}}\exp(-\frac{b}{\sqrt{\epsilon}})),\quad E_2 \approx \epsilon^{-\frac{1}{2}}h_1^2b^4,\quad E_3 \approx h_1^2,$$

$$E_4 \approx \sqrt{\epsilon}h_1(1+\epsilon^{-\frac{1}{2}}\exp(-\frac{b}{\sqrt{\epsilon}})),\quad E_5 \approx \sqrt{\epsilon}h_3.$$

It follows that the accuracy is best when $b \approx \sqrt{\epsilon}\log\epsilon^{-1}$. The error is then

$$\|e\|_\infty \approx \epsilon\delta^2. \quad (3.27)$$

■

We then have the following convergence property for the iterative scheme (3.21),

Lemme 5 *Let B be the interface position defined as in Lemma 4. Suppose that $N^{-1} \approx \epsilon^{\frac{1}{2}}\delta$, with $\delta \succ\succ 1$.*
Then the amplification factor of the iterative scheme is:

$$\xi \approx \delta^{-1}.$$

Preuve: We only need to look at the following homogeneous problem,

$$\left\{\begin{array}{l} L_1^{h_1} e_1^p = 0 \ in \ \Omega_1, \\ e_{1,1}^p = e_{0,1}^p \ , \ e_{N,1}^p = e_{0,2}^p, \\ L_2^{h_3} e_3^p = 0 \ in \ \Omega_3, \\ e_{0,3}^p = e_{N,2}^p \ , \ e_{N,3}^p = 0, \\ L_1^{h_2} e_2^{p+1} = 0 \ in \ \Omega_2, \\ \dfrac{e_{1,2}^{p+1} - e_{0,2}^{p+1}}{h_2} = \dfrac{e_{N,1}^p - e_{N-1,1}^p}{h_1}, \\ \dfrac{e_{N,2}^{p+1} - e_{N-1,2}^{p+1}}{h_2} = \dfrac{e_{1,3}^p - e_{0,3}^p}{h_3}. \end{array}\right. \tag{3.28}$$

We obtain for the first subdomain :

$$e_{i,1}^p = e_{N,1}^p \frac{R^{i-1} + R^{-i}}{R^{N-1} + R^{-N}}, \ \forall i,$$

with $R = 1 + \frac{h_1^2}{2\epsilon} + \frac{h_1}{\sqrt{\epsilon}}\sqrt{1 + \frac{h_1^2}{2\epsilon}} \approx \delta^2$. We have then

$$e_{i,1}^p \prec\prec e_{N,1}^p, \ \forall i < N.$$

We have for the third subdomain :

$$e_{i,3}^p = \frac{N-i}{N} e_{0,3}^p,$$

and then $e_{i,3}^p \leq e_{0,3}$. We have for the second subdomain :

$$e_{i,2}^p = h_2 \frac{e_{0,2}^p}{h_1} \left(\frac{R_*^{i-N+1}}{(R_* - 1)(R_*^{-N+1} - R_*^{N-1})} + \frac{R_*^{N-1-i}}{(R_*^{-1} - 1)(R_*^{N-1} - R_*^{-N+1})} \right)$$

$$-h_2 e_{0,3}^p \left(\frac{R_*^{i}}{(R_* - 1)(R_*^{-N+1} - R_*^{N-1})} + \frac{R_*^{-i}}{(R_*^{-1} - 1)(R_*^{N-1} - R_*^{-N+1})} \right),$$

where $R_* = 1 + \frac{h_2^2}{2\epsilon} + \frac{h_2}{\sqrt{\epsilon}}\sqrt{1 + \frac{h_2^2}{2\epsilon}}$. Using $R_* - 1 \approx \frac{h_2}{\sqrt{\epsilon}}$, we obtain then

$$e_{0,2}^{p+1} \approx \delta^{-1} e_{0,2}^p + 2\sqrt{\epsilon} \exp(-\frac{b}{\sqrt{\epsilon}}) e_{N,2}^p, \ and \ e_{N,2}^{p+1} \approx \sqrt{\epsilon} e_{N,2}^p + 2\sqrt{\epsilon} \exp(-\frac{b}{\sqrt{\epsilon}}) \frac{e_{0,2}^p}{h_1}.$$

We conclude that the amplification factor of the method is then asymptotically δ^{-1}.

■

Combining Lemma 4 and Lemma 5 :

Théorème 5 *With the notations defined above,*

$$\|\phi - \phi^p\|_\infty, \|\psi - \psi^p\|_\infty \leq C(\xi^p + \epsilon\delta^2), \tag{3.29}$$

with

$$\xi \sim \delta^{-1}.$$

Preuve: The proof is a straightforward application of Lemma 4 and Lemma 5. ■

Second scheme :

In this scheme, we adopted the F.Q procedure with the Dirichlet-Neumann boundary conditions to resolve the layer and the transmission problem.

$$\begin{cases} L_1^{h_1}\phi_1^p = F \text{ in } \Omega_1, \\ \phi_1^p(0) = \phi_1^p(h_1)\ ,\ \phi_1^p(B) = \phi_2^p(B), \\ L_2^{h_3}\psi^p = G \text{ in } \Omega_3, \\ \dfrac{\phi_3^p(A+h_3) - \phi^p(A)}{h_3} = \dfrac{\phi_2^p(A) - \phi_2^p(A-h_2)}{h_2}, \\ \psi_3^p(1) = 0, \\ L_1^{h_2}\phi_2^{p+1} = F \text{ in } \Omega_2, \\ \dfrac{\phi_2^{p+1}(B+h_2) - \phi_2^{p+1}(B)}{h_2} = \dfrac{\phi_1^p(B) - \phi_1^p(B-h_1)}{h_1}, \\ \phi_2^{p+1}(A) = \psi_3^p(A). \end{cases} \tag{3.30}$$

With the same principle as in the previous section, we prove the following lemma.

Lemme 6 *Let* $(\tilde{\phi},\tilde{\psi})$ *with* $\tilde{\phi} = (\tilde{\phi}_{i,j})_{i=0,\dots N, j=1,2}$ *and* $\tilde{\psi} = (\tilde{\psi}_i)_{i=0,\dots N}$ *be the solution of the linear system obtained as the formal limit of (3.30) when* $p \to \infty$.

Let M *be the composite grid defined above. Let us suppose that* $N^{-1} \approx \sqrt{\epsilon}\delta$, *with* $\delta \succ\succ 1$. *Let* $\| \ \|_\infty$ *be the maximum norm on the composite grid* M.

Under the previous hypothesis on the discretization and approximation of the operators in each subdomain, $\max(\| \phi - \tilde{\phi} \|_\infty, \| \psi - \tilde{\psi} \|_\infty)$ *is asymptotically minimum when*

$$b = A - B \approx \sqrt{\epsilon}\log(\epsilon^{-1}).$$

We then have the following property of the iterative scheme.

Lemme 7 *Let* B *be the interface position defined above. Suppose that* $N^{-1} \approx \sqrt{\epsilon} \times \delta$ *with* $\delta \gg 1$. *Then the iterative scheme cannot converge.*

Proof : We look at the following homogeneous problem,

$$\begin{cases} L_1^{h_1} e_1^p = 0 \text{ in } \Omega_1, \\ e_{1,1}^p = e_{0,1}^p\ ,\ e_{N,1}^p = e_{0,2}^p, \\ L_2^{h_3} e_3^p = 0 \text{ in } \Omega_3, \\ \dfrac{e_{1,3}^p - e_{0,3}^p}{h_3} = \dfrac{e_{N,2}^p - e_{N-1,2}^p}{h_2}, \\ e_{N,3}^p = 0, \\ L_1^{h_2} e_2^{p+1} = 0 \text{ in } \Omega_2, \\ \dfrac{e_{1,2}^{p+1} - e_{0,2}^{p+1}}{h_2} = \dfrac{e_{N,1}^p - e_{N-1,1}^p}{h_1}, \\ e_{N,2}^{p+1} = e_{0,3}^p. \end{cases} \tag{3.31}$$

As for the previous model, we can easily prove :

$e^p_{i,1} \prec\prec e^p_{N,1}$, $\forall i < N$ and $e^p_{i,3} \leq e^p_{0,3}$ and we have

$$\begin{cases} e^{p+1}_{0,2} \approx \frac{\sqrt{\epsilon}}{h_1} e^p_{0,2} + \exp\left(-\frac{b}{\sqrt{\epsilon}}\right) e^p_{N,2}, \\ e^{p+1}_{N,2} = e^p_{N,2}. \end{cases}$$

Since the error at the point A of Ω_2 stays constant, the scheme does not converge.

Third scheme:

Let us use now the *Schwarz alternate procedure* to solve the layer. We keep the F.Q scheme with N - D boundary conditions, to solve the transmission condition in A.
We restrict ourselves to an overlap minimum i.e one cell of step h, between $\Omega_1 = [0,a]$ and $\Omega_2 = [b,1]$ $(with \quad 0 < b < a < 1)$.
Furthermore, to simplify the proof, we impose that the grids of the subdomains Ω_1 and Ω_2 coincide at the boundary points. Assume that $m \times h_2 = h_1$.
The iterative scheme writes:

$$\begin{cases} L_1^{h_1} \phi_1^p = F \ in \ \Omega_1, \\ \phi_1^p(0) = \phi_1^p(h_1) \ , \ \phi_1^p(a) = \phi_2^p(b + m * h_2), \\ L_2^{h_3} \psi^p = G \ in \ \Omega_3, \\ \phi_3^p(A) = \phi_2^p(A) \ , \ \psi_3^p(1) = 0, \\ L_1^{h_2} \phi_2^{p+1} = F \ in \ \Omega_2, \\ \dfrac{\phi_2^{p+1}(A) - \phi_2^{p+1}(A - h_2)}{h_2} = \dfrac{\psi^p(A + h_3) - \psi^p(A)}{h_3}. \end{cases} \tag{3.32}$$

It can be proved that $\max(\| \phi - \tilde{\phi}\|_\infty, \| \psi - \tilde{\psi}\|_\infty)$ is asymptotically minimum when

$$A - b \approx \sqrt{\epsilon} \log(\epsilon^{-1}).$$

We then have the following convergence property of the iterative scheme.
Lemme 8 *Let B be the interface position defined above. Suppose that* $N^{-1} \approx \epsilon^{\frac{1}{2}} \delta$, *with* $\delta \succ\succ 1$. *Then the amplification factor of the iterative scheme is:*

$$\xi \approx \sqrt{\epsilon} \log(\epsilon^{-1}) \delta.$$

Preuve: We look at the following homogeneous problem,

$$\begin{cases} L_1^{h_1} e_1^p = 0 \ in \ \Omega_1, \\ e^p_{1,1} = e^p_{0,1} \ , \ e^p_{N,1} = e^p_{m,2}, \\ L_2^{h_3} e_3^p = 0 \ in \ \Omega_3, \\ e^p_{0,3} = e^p_{N,2} \ , \ e^p_{N,3} = 0, \\ L_1^{h_2} e_2^{p+1} = 0 \ in \ \Omega_2, \\ e^{p+1}_{0,2} = e^p_{N-1,1}, \\ \dfrac{e^{p+1}_{N,2} - e^{p+1}_{N-1,2}}{h_2} = \dfrac{e^p_{1,3} - e^p_{0,3}}{h_3}. \end{cases} \tag{3.33}$$

Using the same principle as for lemma5, we obtain:

- For the first and the third subdomain error
  $e^p_{i,1} \prec\prec e^p_{N,1}$ and $e^p_{i,3} \leq e_{0,3} \quad \forall i < N,$
- for the second subdomain error
  $e^{p+1}_{0,2} \approx \epsilon e^p_{m,2}$ *and* $e^{p+1}_{N,2} \approx \epsilon \log(\epsilon^{-1}) \delta e^p_{N,2} + \epsilon^2 e^p_{m,2}.$
  We have: $e^p_{m,2} \approx R_*^{m-N} e^p_{N,2} + R_*^{-m} e^p_{0,2}$ with $R_* = 1 + \frac{h_2^2}{2\epsilon} + \frac{h_2}{\sqrt{\epsilon}}\sqrt{1 + \frac{h_2^2}{2\epsilon}}.$

We easily conclude that the amplificator factor of the method is asymptotically $\delta\epsilon \log(\epsilon^{-1})$. ■

Finally we have:

Théorème 6 *With the notations defined above, applying the F.Q and Schwarz mixed method, we have*

$$\max(\|\phi - \phi^p\|_\infty, \|\psi - \psi^p\|_\infty) \leq C(\xi^p + \epsilon\delta^2), \tag{3.34}$$

with

$$\xi \sim \delta\epsilon \log \epsilon^{-1}.$$

We have implemented each of these schemes on the model problem (3.17) with F=0 and $G = \exp(-20 \times (x - 0.8)^2)$. The criterion to stop the scheme is that the jump at the interface A between two consecutive iterates is less than 10^{-4}. For a fixed N, the first and third schemes approximate the solution with the same order of accuracy. The following table of results is in agreement with our analysis. It shows in particular that the third scheme is the most efficient.

Number of iterations	$\epsilon = 10^{-3}$ N=5	$\epsilon = 10^{-3}$ N=10	$\epsilon = 10^{-3}$ N=20	$\epsilon = 10^{-3}$ N=30	$\epsilon = 10^{-4}$ N=10	$\epsilon = 10^{-4}$ N=20
First scheme	10	14	23	33	6	8
Second scheme	DIV	DIV	DIV	DIV	DIV	DIV
Third scheme	3	4	5	8	3	3

Number of iterations	$\epsilon = 10^{-4}$ N=30	$\epsilon = 10^{-4}$ N=40	$\epsilon = 10^{-5}$ N=10	$\epsilon = 10^{-5}$ N=20	$\epsilon = 10^{-5}$ N=30	$\epsilon = 10^{-5}$ N=40
First scheme	10	12	5	5	6	6
Second scheme	DIV	DIV	DIV	DIV	DIV	DIV
Third scheme	4	5	3	3	3	3

3.3 Boundary layers in a Two Dimensional Space

Let us consider the analogous of (3.1) in a two dimensional space, that is:

$$\begin{cases} L_\epsilon \phi = -\epsilon\Delta\phi + \gamma\phi = F \text{ in } \Omega, \\ \phi = g \text{ on } \partial\Omega, \end{cases} \tag{3.35}$$

where Ω is a disc of radius 1. We write this problem in polar coodinates, with $\Omega = \{(r,\theta) \in [0,1[\times[0,2\pi]\}$. We assume that γ and F are differentiable functions with respect to r and θ, independent of ϵ, with uniformly bounded derivatives on $\bar{\Omega}$. Where ϵ is a small positive parameter,$\epsilon \in]0,\epsilon_0]$; $\epsilon_0 > 0$, γ is strictly positive such that:

$$\gamma(r,\theta) \geq \gamma_0 > 0 \ (r,\theta) \in \Omega,$$

and ϕ has a boundary layer in the neighborhood of $\partial\Omega$ of $\sqrt{\epsilon}$ thichness([23]). The domain Ω is split into two subdomains:

$$\begin{aligned} \Omega_{inner} &= [a,1] \times [0,2\pi] \\ \Omega_{outer} &= [0,a] \times [0,2\pi] \ \text{ where } a > 0. \end{aligned}$$

On each subdomain, the differential expressions are discretized by means of a finite difference scheme. We approximate the Laplacian operator by the five-point scheme. We assume the mesh to be regular in polar coordinates in both subdomains.
Let (h_1,h_θ)(respt (h_2,h_θ) be the mesh sizes in Ω_{outer} (respt Ω_{inner}). We assume (3.2) for the space step in the radial direction. The goal of this section is to show how to generalize the results in one space dimension to two space dimensions by means of comparison lemma. For the sake of simplicity, we will restrict ourselves to the Dirichlet - Neumann scheme analogue to (2.3), that is:

$$\begin{cases} L^{h_1}\phi^p_{outer} = F \ on \ \Omega_{outer}, \\ \phi^p_{outer}(1,.) = \phi^p_{outer}(0,.) \ , \ \phi^p_{outer}(a,.) = \phi^p_{inner}(a,.), \\ L^{h_2}\phi^{p+1}_{inner} = F \ on \ \Omega_{inner}, \\ \phi^{p+1}_{inner}(0,.) = g(\theta), \\ \dfrac{\phi^{p+1}_{inner}(a+h_2,.) - \phi^{p+1}_{inner}(a,.)}{h_2} = \dfrac{\phi^p_{outer}(a,.) - \phi^p_{outer}(a-h_1,.)}{h_1}. \end{cases}$$

To start the scheme, we impose an artificial boundary condition at the circle of radius a. We define the interface position, i.e the radius a, to be such that the maximum of the truncation error in each subdomain is asymptotically minimum. This interface position is optimal when the error at the interface goes to zero. The truncation error for the outer domain is:

$$e_{outer} \sim C \max\left(h_{r_1}^2 a^2 \max_{\Omega_{outer}}\left(\left|\frac{d^{(3)}\phi}{dr^3}\right|, \left|\frac{d^{(4)}\phi}{dr^4}\right|\right), h_y^2 \max\left|\frac{d^{(4)}\phi}{dy^4}\right|\right).$$

The truncation error for the outer domain is:

$$e_{inner} \sim C \max\left(h_{r_2}(1-a) \max_{\Omega_{inner}}\left|\frac{d^{(2)}\phi}{dr^2}\right|, h_y^2 \max\left|\frac{d^{(4)}\phi}{dr^4}\right|\right).$$

We look now for a, such that $\max(e_{outer},e_{inner})$ is minimum. The interface position defined above depends only on the property of the solution that we want to approximate.
Let ϕ_0 be the outer expansion of ϕ. $\Theta(r,\theta,\epsilon)$ be the corrector, that is:

$$\Theta(r,\theta,\epsilon) = \phi(r,\theta,\epsilon) - \phi_0(r,\theta,\epsilon) \text{ in the boundary layer.}$$

It can be shown as in [29] that :

Lemme 9 *Suppose that*

(i) the corrector Θ is strictly of order one,

(ii) the inner variable $(\xi,\theta) = (\frac{1-r}{\sqrt{\epsilon}},\theta)$ and all derivatives of $\tilde{\Theta}(\xi,\theta,\epsilon) = \Theta(r,\theta,\epsilon)$ with respect to ξ are of order one,

(iii) the corrector function ξ is exponentially decreasing $\Theta(\xi,\theta,\epsilon) \sim C\exp(-C_0\xi)$ as $\xi \to +\infty$.

Then the accuracy is optimal in both subdomain if

$$1 - a \approx \sqrt{\epsilon}\log(\epsilon^{-1}).$$

The main task is to extend the results of section 2.1.1 with some comparison lemma.

Lemme 10 *Take $(\phi_{i,j})_{\substack{i=0,\dots N\\ j=0,\dots N}}$ solution of the following discretized 2D-Elliptic problem :*

$$\begin{cases} -\epsilon\left(\dfrac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{h^2} + \dfrac{1}{(ih)^2}\dfrac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{h_\theta^2} + \dfrac{1}{ih}\dfrac{\phi_{i+1,j} - \phi_{i,j}}{h}\right) + \gamma_{i,j}\phi_{i,j} = 0 \\ \qquad\qquad \forall i = 1,\dots N-1 \;\; \forall j = 1,\dots N-1, \\ \phi_{1,j} = \phi_{0,j} \qquad \forall j = 0,\dots N, \\ \phi_{N,j} = \alpha(j) \qquad \forall j = 0,\dots N, \\ \phi(i,.) \qquad \forall i = 0,\dots N-1 \qquad \textit{periodic.} \end{cases}$$

Let $(e_i)_{i=0,\dots N}$ be the solution of the following discretized 1D problem :

$$\begin{cases} -\epsilon\dfrac{2e_{i+1} - 3e_i + e_{i-1}}{h^2} + \gamma_0 e_i = 0 \quad \forall i = 1,\dots N-1, \\ e_N = \alpha_0, \\ e_1 = e_0, \end{cases}$$

with

$$\gamma_{i,j} \geq \gamma^0 > 0 \quad \forall i,j = 0,\dots N,\ \alpha^0 \geq \max_{j=0,\dots N}|\alpha(j)|. \tag{3.36}$$

Then

$$|\phi_{i,j}| \leq e_i \qquad \forall i,j = 0,\dots N.$$

Preuve:

$\circ$ $E_i = \phi_{i,j_0(i)} = \max\limits_{j=0,\dots N}(\phi_{i,j})$ satisfies

$$\begin{cases} -\epsilon\left(\dfrac{E_{i+1} - 2E_i + E_{i-1}}{h^2} + \dfrac{1}{ih}\dfrac{E_{i+1} - E_i}{h}\right) + \gamma_{i,j_0(i)}E_i \leq 0 \quad \forall i = 1,\dots N-1, \\ E_N = \max\limits_{j=0,\dots N}(\alpha(j)) = \beta \quad , \quad E_1 = E_0. \end{cases} \tag{3.37}$$

Let us denote

$$\tilde{\gamma}_i = \gamma_{i,j_0(i)},\ a_i = \frac{2\epsilon}{h^2} + \frac{\epsilon}{ih^2} + \tilde{\gamma}_i,\ b_i = \frac{\epsilon}{h^2} + \frac{\epsilon}{ih^2},\ c_i = \frac{\epsilon}{h^2}.$$

(3.37) rewrites:

$$a_i E_i \leq b_i E_{i+1} + c_i E_{i-1},\ E_N = \beta,\ E_1 = E_0,$$

with $a_i > b_i + c_i \quad a_i > 0\ ,\ b_i > 0\ ,\ c_i > 0.$

We show first that:

$$E_i \leq \bar{E}_i \qquad \forall i = 0, \ldots N \tag{3.38}$$

where

$$a_i \bar{E}_i = b_i \bar{E}_{i+1} + c_i \bar{E}_{i-1},\ \bar{E}_N = \alpha^0 \geq |\beta|,\ \bar{E}_1 = \bar{E}_0.$$

∘ Now let $(\hat{e}_i)_{i=0,\ldots N}$ be the solution of the following problem:

$$\begin{cases} -\epsilon\left(\dfrac{\hat{e}_{i+1} - 2\hat{e}_i + \hat{e}_{i-1}}{h^2} + \dfrac{1}{ih}\dfrac{\hat{e}_{i+1} - \hat{e}_i}{h}\right) + \gamma^0 \hat{e}_i = 0 \quad \forall i = 1, \ldots N-1 \\ \hat{e}_N = \alpha^0 \\ \hat{e}_1 = \hat{e}_0. \end{cases}$$

Let us suppose that $\hat{e}_0 < 0$; then from the equality:

$$\epsilon\left(\frac{1}{h^2} + \frac{1}{ih^2}\right)(\hat{e}_{i+1} - \hat{e}_i) = \frac{\epsilon}{h^2}(\hat{e}_i - \hat{e}_{i-1}) + \gamma^0 \hat{e}_i \quad \forall i = 1, \ldots N-1,$$

we conclude by finite induction that $(\hat{e}_i)_{i=0,\ldots N}$ decreases and stays strictly negative, this is in contradiction with $\hat{e}_N = \alpha^0 \geq 0$.

We therefore have $\hat{e}_0 \geq 0$ and show by finite induction that $\hat{e}_i$ is increasing and stay positive. Now let us compare $(\hat{e}_i)$ and $(\bar{E}_i)$; $e_i^\star = \bar{E}_i - \hat{e}_i$ satisfies:

$$\begin{cases} -\epsilon\left(\dfrac{e^\star_{i+1} - 2e^\star_i + e^\star_{i-1}}{h^2} + \dfrac{1}{ih}\dfrac{e^\star_{i+1} - e^\star_i}{h}\right) + \gamma^0 e^\star_i + (\tilde{\gamma}_i - \gamma^0)\hat{e}_i = 0 \quad \forall i = 1, \ldots N-1 \\ e^\star_N = 0 \\ e^\star_1 = e^\star_0. \end{cases}$$

Take $a_i^0 = \dfrac{2\epsilon}{h^2} + \dfrac{\epsilon}{ih^2} + \gamma^0$ $(e_i^\star)$ satisfies

$$a_i^0 e_i^\star \leq b_i e^\star_{i+1} + c_i e^\star_{i-1} \qquad \forall i = 1, \ldots N-1,\ e^\star_N = 0,\ e^\star_1 = e^\star_0.$$

Since $a_i^0 > b_i + c_i$, we have

$$\bar{E}_i \leq \hat{e}_i \qquad \forall i = 0, \ldots N. \tag{3.39}$$

∘ Because $(\hat{e}_i)$ is increasing, we have

$$\begin{cases} -\epsilon\dfrac{2\hat{e}_{i+1}-3\hat{e}_i+\hat{e}_{i-1}}{h^2}+\gamma^0\hat{e}_i\leq 0 & \forall i=0,\dots N-1 \\ \hat{e}_N=\alpha^0 \\ \hat{e}_1=\hat{e}_0. \end{cases}$$

We notice that

$$\hat{e}\leq e_i \qquad \forall i=0,\dots N. \tag{3.40}$$

with (e_i) solution of:

$$\begin{cases} -\epsilon\dfrac{2e_{i+1}-3e_i+e_{i-1}}{h^2}+\gamma^0 e_i=0 & \forall i=1,\dots N-1 \\ e_N=\alpha^0\geq \max\limits_{j=0,\dots N}|\alpha(j)| \\ e_1=e_0. \end{cases}$$

We conclude using (3.38), (3.39), (3.40) that: $\phi_{i,j}\leq e_i \qquad \forall i=0,\dots N.$

With an entirely similar analysis, we have also: $-\phi_{i,j}\leq e_i \qquad \forall i=0,\dots N.$

■

Lemme 11 *Let* $(\phi_{i,j})_{\substack{i=0,\dots N\\ j=0,\dots N}}$ *be the solution of the following discretized 2D-Elliptic problem:*

$$\begin{cases} -\epsilon\left(\dfrac{\phi_{i+1,j}-2\phi_{i,j}+\phi_{i-1,j}}{h^2}+\dfrac{1}{(i(h+a))^2}\dfrac{\phi_{i,j+1}-2\phi_{i,j}+\phi_{i,j-1}}{h_\theta^2}+\dfrac{1}{i(h+a)}\dfrac{\phi_{i+1,j}-\phi_{i,j}}{h}\right)+\gamma_{i,j}\phi_{i,j}=0 \\ \qquad\qquad \forall i=1,\dots N-1 \ \ \forall j=1,\dots N-1, \\ \dfrac{\phi_{1,j}-\phi_{0,j}}{h}=\alpha(j) \qquad \forall j=0,\dots N, \\ \phi(N,j)=0 \qquad \forall j=0,\dots N-1. \end{cases}$$

Let $(e_i)_{i=0,\dots N}$ *be the solution of the following discretized 1D problem:*

$$\begin{cases} -\epsilon\dfrac{2e_{i+1}-3e_i+e_{i-1}}{h^2}+\gamma_0 e_i=0 \quad \forall i=1,\dots N-1, \\ \dfrac{e_1-e_0}{h}=-\alpha^0, \\ e_N=0, \end{cases}$$

with

$$\gamma_{i,j}\geq\gamma^0>0 \quad \forall i,j=0,\dots N, \alpha^0\geq\max_{j=0,\dots N}|\alpha(j)|.$$

Then

$$|\phi_{i,j}|\leq e_i \qquad \forall i,j=0,\dots N.$$

Preuve: This proof is very similar to the proof of the previous lemma.
∘ Let $E_i = \phi_{i,j_0(i)} = \max_{j=0,\dots N}(\phi_{i,j})$.
We have:

$$\begin{cases} -\epsilon\left(\dfrac{E_{i+1}-2E_i+E_{i-1}}{h^2}+\dfrac{1}{ih+a}\dfrac{E_{i+1}-E_i}{h}\right)+\tilde{\gamma}_i E_i \le 0 \quad \forall i=1,\dots N-1, \\ E_N = 0, \\ \dfrac{E_1-E_0}{h} \ge -\alpha^0. \end{cases}$$

Now, take $(\hat{e}_i)_{i=0,\dots N}$ be the solution of the following problem:

$$\begin{cases} -\epsilon\left(\dfrac{\hat{e}_{i+1}-2\hat{e}_i+\hat{e}_{i-1}}{h^2}+\dfrac{1}{ih+a}\dfrac{\hat{e}_{i+1}-\hat{e}_i}{h}\right)+\gamma^0\hat{e}_i = 0 \quad \forall i=1,\dots N-1, \\ \hat{e}_N = 0, \\ \dfrac{\hat{e}_1-\hat{e}_0}{h} = -\alpha^0. \end{cases}$$

Let us suppose that $\hat{e}_{N-1} < 0$; then from the equality:

$$\epsilon\left(\frac{1}{h^2}+\frac{1}{ih^2}\right)(\hat{e}_{i+1}-\hat{e}_i) = \frac{\epsilon}{h^2}(\hat{e}_i-\hat{e}_{i-1})-\gamma^0\hat{e}_i \quad \forall i=1,\dots N-1.$$

We conclude by backward finite induction that $\hat{e}_i$ is strictly increasing. This is in contradiction with $\hat{e}_1-\hat{e}_0 = -\alpha^0 h$.
We therefore have $\hat{e}_{N-1} \ge 0$ and conclude by finite induction that $\hat{e}_i$ is decreasing and stay positive. Now let us compare $(\hat{e}_i)$ and (E_i). Let $\tilde{e}_i = E_i - \hat{e}_i$, it satisfies:

$$\begin{cases} -\epsilon\left(\dfrac{\tilde{e}_{i+1}-2\tilde{e}_i+\tilde{e}_{i-1}}{h^2}+\dfrac{1}{ih+a}\dfrac{\tilde{e}_{i+1}-\tilde{e}_i}{h}\right)+\tilde{\gamma}\tilde{e}_i+(\tilde{\gamma}_i-\gamma^0)\hat{e}_i \le 0 \quad \forall i=1,\dots N-1, \\ \tilde{e}_N = 0, \\ \tilde{e}_1-\tilde{e}_0 \ge 0. \end{cases}$$

Let us denote $\forall i = 1,\dots N-1$

$$a^i = \epsilon\left(\frac{2}{h^2}+\frac{1}{h(ih+a)}\right)+\tilde{\gamma}_i,\ b_i = \epsilon\frac{1}{h^2}+\frac{1}{h(ih+a)} \quad , \quad c_i = \epsilon\frac{1}{h^2}.$$

We have:

$$a_i\tilde{e}_i \le b_i\tilde{e}_{i+1}+c_i\tilde{e}_{i-1} \qquad \forall i=1,\dots N-1,\ \tilde{e}_N = 0,\ \tilde{e}_1 \ge \tilde{e}_0.$$

With $a_i^0 > b_i + c_i \qquad \forall i = 1,\dots N-1$, we conclude then:

$$\tilde{e}_i \le 0 \qquad \forall i=0,\dots N, \text{ and } \bar{E}_i \le \hat{e}_i \qquad \forall i=0,\dots N.$$

o Because $(\hat{e}_i)$ is decreasing, we have at last:

$$\begin{cases} -\epsilon\dfrac{\hat{e}_{i+1}-2\hat{e}_i+\hat{e}_{i-1}}{h^2}+\gamma^0\hat{e}_i\le 0 & \forall i=0,\dots N-1,\\ \hat{e}_N=0,\\ \hat{e}_1-\hat{e}_0=-\alpha^0 h. \end{cases}$$

It is straightforward to show that:

$$\hat{e}\le e_i \qquad \forall i=0,\dots N,$$

where (e_i) is the solution of:

$$\begin{cases} -\epsilon\dfrac{e_{i+1}-2e_i+e_{i-1}}{h^2}+\gamma^0 e_i=0 & \forall i=1,\dots N-1\\ e_N=0\\ e_1-e_0=-\alpha^0 h. \end{cases}$$

We conclude that: $\phi_{i,j}\le e_i \qquad \forall i=0,\dots N$

The same analysis holds for $-\phi_{i,j}$.

■

Using the previous two lemmas, we can now obtain the following convergence estimate:

Théorème 7 *Let the circle with the equation $r=a$, where $1-a\approx\sqrt{\epsilon}\log\epsilon^{-1}$, as being the interface position, with ξ the amplificator factor.*
If $N_r\approx\epsilon^{1/2}\delta$ with $\delta\gg 1$ then

$$\xi\approx\delta^{-1}.$$

Preuve: Let $\tilde{\phi}=(\tilde{\phi}^k_{i,j})_{\substack{i=0,\dots N;j=0,\dots N\\ k=1,2}}$ be the solution of the following system

$$\begin{cases} L_1^{h_1}\tilde{\phi}^1_{i,j}=F^1_{i,j} & i=1,\dots N-1\ ;\quad j=1,\dots N-1,\\ \tilde{\phi}^1_{1,j}=\phi^1_{0,j}\ ,\quad \tilde{\phi}^1_{N,j}=\tilde{\phi}^2_{0,j} & j=0,\dots N,\\ L_2^{h_2}\tilde{\phi}^2_{i,j}=F^2_{i,j} & i=1,\dots N-1\ ;\ j=1,\dots N-1,\\ \tilde{\phi}^2_{N,j}=g_j & j=0,\dots N,\\ \dfrac{\tilde{\phi}^2_{1,j}-\tilde{\phi}^2_{0,j}}{h_2}=\dfrac{\tilde{\phi}^1_{N,j}-\tilde{\phi}^1_{N-1,j}}{h_1}\ ; & j=0,\dots N, \end{cases}$$

where

$$L_k^h\phi_{i,j}=-\epsilon\left(\frac{\phi_{i+1,j}-2\phi_{i,j}+\phi_{i-1,j}}{h^2}+\frac{1}{r_k(i)}\frac{\phi_{i,j}-\phi_{i-1,j}}{h}+\frac{1}{r_k(i)^2}\frac{\phi_{i,j+1}-2\phi_{i,j}+\phi_{i,j-1}}{h_\theta^2}\right)+\gamma_{i,j}\phi_{i,j}$$

with $r_1(i)=ih$ and $r_2(i)=a+ih$.
The existence and uniqueness of $(\tilde{\phi}_{i,j})$ follows from the *maximum principle*. To prove the convergence of the Dirichlet-Neumann scheme, it is sufficient to restrict ourselves to the homogeneous problem ie $F^k_{i,j}=g_j=0 \quad \forall i,j,k$.

Let $\tilde{\phi}_{outer} = (\tilde{\phi}^1_{i,j})_{\substack{i=0,\dots N\\ j=0,\dots N}}$, $e^p_{outer} = \phi^p_{outer} - \tilde{\phi}_{outer}$ (resp $e^p_{inner} = \phi^p_{inner} - \tilde{\phi}_{inner}$).

Let us denote: $e^p_{outer} = (e^{1,p}_{i,j})_{\substack{i=0,\dots N_r\\ j=0,\dots N_\theta}}$ (resp $e^p_{inner} = (e^{2,p}_{i,j})_{\substack{i=0,\dots N_r\\ j=0,\dots N_\theta}}$) and $(e^{k,p}_i)_{\substack{i=0,\dots N\\ k=1,2}}$ such that:

$$\begin{cases} -\epsilon\dfrac{2e^{1,p}_{i+1} - 3e^{1,p}_i + e^{1,p}_{i-1}}{h1^2} + \gamma_0 e^{1,p}_i = 0 \qquad \forall i = 1,\dots N-1, \\ e^{1,p}_1 = e^{1,p}_0 \ , \ e^{1,p}_N = e^{2,p}_0, \\ -\epsilon\dfrac{e^{2,p+1}_{i+1} - 2e^{2,p+1}_i + e^{2,p+1}_{i-1}}{h_2^2} + \gamma_0 e^{2,p+1}_i = 0 \qquad \forall i = 1,\dots N-1, \\ \dfrac{e^{2,p+1}_1 - e^{2,p+1}_0}{h_2} = -\dfrac{e^{1,p}_N + e^{1,p}_{N-1}}{h_1} \quad , \quad e^{2,p+1}_N = 0, \end{cases}$$

with

$$e^{1,p}_N = \max_{j=0,\dots N} |\phi^{1,0}_{i,j}|.$$

From lemma 10, we have:

$$|\phi^{1,0}_{i,j}| \le e^{1,0}_i \qquad \forall i,j = 0,\dots N,$$

and then

$$-\max\left|\frac{\phi^{1,0}_{N,j} - \phi^{1,0}_{N-1,j}}{h_1}\right| \ge -\frac{e^{1,0}_N + e^{1,0}_{N-1}}{h_1}.$$

We can conclude from lemma 11 that:

$$|\phi^{2,1}_{i,j}| \le e^{2,1}_i \qquad \forall i,j = 0,\dots N.$$

By induction, we show that:

$$\begin{cases} |\phi^{1,p}_{i,j}| \le e^{1,p}_i \qquad \forall i,j = 0,\dots N \ ; \ \forall p \ge 1, \\ |\phi^{2,p}_{i,j}| \le e^{2,p}_i \qquad \forall i,j = 0,\dots N \ ; \ \forall p \ge 1. \end{cases}$$

Let us now show that $\max_{j=0,\dots N} |e^{k,p}_i|$ converges to 0, $\forall k = 1,2$.

We use the notation $\hat{\epsilon} = \frac{\epsilon}{\gamma^0}$. An explicit calculus gives: $e^{1,p}_{N-1} \sim 2\frac{\hat{\epsilon}}{h^2} e^{1,p}_N$ and therefore:

$$e^{1,p}_{N-1} \ll e^{1,p}_N.$$

Consequently we have:

$$\begin{aligned} e^{2,p+1}_1 - e^{2,p+1}_0 &\approx \tfrac{h_2}{h_1} e^{1,p}_N \\ &\approx \tfrac{h_2}{h_1} e^{2,p}_0. \end{aligned}$$

We conclude as in the one dimensional case that the damping factor is given by:

$$\xi \approx \frac{h_2}{h_1} \times \frac{1 - R^{2N+1}}{1 + R^{2N-1}} \times \frac{1}{R-1},$$

where R is the larger root of the quadratic polynomial:

$$R^2 - (2 + \frac{h_2^2}{\hat{\epsilon}})R + 1 = 0.$$

Finally, we have

$$\xi \approx \delta^{-1}.$$

■

3.4 Applications

3.4.1 Problème de transmission simplifié

Our objective is to solve efficiently a two dimensional singular perturbed transmission problem that arises in electro-magnetic theory([16]) by using the *Modulef* finite element package. To start, we consider the following model:

$$\begin{cases} -\epsilon\Delta u + a(r,\theta)u = 0 & in \quad \Omega^1 \\ -\Delta u = j(r,\theta) & in \quad \Omega^2, \\ [u] = [\frac{\partial u}{\partial n}] = 0 & on \quad \partial\Omega^1 \cap \partial\Omega^2 \\ u(R_\infty,\theta) = 0 & for \quad \theta \in (0,2\pi). \end{cases} \tag{3.41}$$

Here Ω_1 is a disk of radius one, Ω_2 is a ring for $r \in (1,R_\infty)$, j represents a given current density in the inductor, Ω^1 is the domain of the solid conductor, Ω^2 is the exterior domain of the conductor truncated by (R^∞,θ) and the boundary layer in Ω^1 corresponds to the well-known skin effect. For the numerical resolution, we arbitrarily fixed R^∞ at 20.
This simplified model is only a small part of the real magnetic model. However our first experiments with the F.Q iterative procedure in solving alternatively the equations in Ω_1 and Ω_2 did show a problem, where convergence of the algorithm strongly depended on the radius of the elements of the unstructured grid used as well as on ϵ. To be more specific, the iterative procedure converged slowly and could even diverge if the mesh was not appropriate. Applying the **comparison lemma** as previously (Theorem 7), we extend our results obtained for heterogeneous domain decomposition from a one dimensional space to a two dimensional space. We have therefore modified our implementation according to the analysis in the following way : we use the "F.Q. and Schwarz mixed method" analogous to the third scheme of Sect 2.2. The main advantage of this approach is that it is easy to use the *Modulef* library to follow this iterative procedure. We used three subdomains, two located on the metal (Ω^1_{outer} and Ω^1_{inner}) and an external subdomain (Ω_2). For each subdomain we used an automatic mesh generator governed by the algebraic method of a generalized triangle for the mesh of a disc and that of a generalized quadrilateral for the mesh of a ring, for which we have given the boundary descriptions and their discretization. We made a regular discretization along the radius of each subdomain such that the number of finite elements is roughly of the same order in each subdomain. We have then applied the hypothesis of Theorem 7 about the discretization (cf figures 3.5, 3.6 and 3.7)

as follows:
let h_1(respectively h_2) be the discretization step along the radius of Ω^1_{outer}(respectively Ω^1_{inner}).
We have the following hypothesis about a,b,h_1, h_2:

$$\begin{cases} a - b = h_1, \\ 1 - b \approx \sqrt{\epsilon} \log \epsilon^{-1}. \end{cases}$$

The variational expression of an iteration of our algorithm is the following:

Find u_1^p,u_2^p,u_3^p such that:

- $u_1^p \in u_2^{p-1}/\partial\Omega^1_{outer} + H_0^1(\Omega^1_{outer})$

  and
  $$\epsilon \int_{\Omega^1_{outer}} \nabla u_1^p . \nabla v + \int_{\Omega^1_{outer}} u_1^p . v = 0 \qquad \forall v \in H_0^1(\Omega^1_{outer}),$$

- $u_3^p \in u_2^{p-1}/(\partial\Omega^1_{inner} \cap \Omega^2) + H_0^1(\Omega^2)$

  and
  $$\int_{\Omega^2} \nabla u_3^p . \nabla v = \int_{\Omega^2} j.v \qquad \forall v \in H_0^1(\Omega^2),$$

- $u_2^p \in u_1^p/\partial\Omega^1_{outer} + V_{inner}$ with $V_{inner} = \{v \in H^1(\Omega^1_{inner}) \ ; \ v/\partial\Omega^1_{outer} = 0\}$

  and
  $$\epsilon \int_{\Omega^1_{inner}} \nabla u_2^p . \nabla v + \int_{\Omega^1_{inner}} u_2^p . v = \epsilon \int_\Gamma \frac{\partial u}{\partial n} v \qquad \forall v \in V^1_{inner}.$$

We used for current density $j(r,\theta) = \sin(2\theta + \frac{3\pi}{4}) \times \exp(-\frac{\sin(4\theta)+1}{0.1}) \times \exp(-(r-10)^2)$.
We refined the mesh in the neighbourhood of the inductors (cf figure3.7)

To resolve our problem for these meshes, we used *curved finite elements* (triangles with six nodes, *P2-Lagrange-isoparametric*) and a *direct solver (Factorization of Cholesky)*.

In Table 3.1, one can find the results of our numerical experiment; the criterion to stop the iteration is a jump at the interfaces between Ω^1_{inner} and Ω^2, Ω^1_{outer} less than 10^{-8}. For several values of N (number of points along the radius for each subdomain), we give the number of iterations and the number of elements for each subdomain.

We have checked that our numerical scheme, when applied to (3.41), converged with the same number of iterations.

Figure 3.9-3.11 show the isovalues of the solution from each subdomain, with N=10 and $\epsilon = 10^{-3}$. We resolved (3.41) on this mesh for different values of ϵ and observed that we must adapt the mesh to obtain a good rate of convergence(cf Table 3.2 row 1).

	N=10 NE1=320 NE2=576 NE3=992	N=15 NE1=780 NE2=1456 NE3=2576	N=20 NE1=1440 NE2=2736 NE3=4752
$\epsilon = 10^{-2}$	7	7	11
$\epsilon = 10^{-3}$	5	4	6
$\epsilon = 10^{-4}$	4	4	4
$\epsilon = 10^{-5}$	3	3	3

TAB. 3.1 – *number of iterations*

	$\epsilon = 1.$	$\epsilon = 0.1$	$\epsilon = 10^{-2}$	$\epsilon = 10^{-3}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-5}$
meshes adapted to N=10 $\epsilon = 10^{-3}$	169	72	22	7	7	6
regular meshes with arbitrary interface N=10	-	-	DIV	DIV	DIV	DIV
regular meshes with optimal interface position for $\epsilon = 10^{-3}$ N=10	-	DIV	16	12	8	7

TAB. 3.2 – *number of iterations*

We have also applied an algorithm to a regular mesh with approximatively the same order of discretization for each subdomain. We consider two cases, a first case with an arbitrary interface position and a second case with an interface position adapted for $\epsilon = 10^{-3}$ (cf Table 3.2 rows 2, 3).

We observe that since this computation was done with finite element framework with *Modulef*, it could be extended to more general geometries.

3.4.2 Stationary case

We have shown in the previous chapter that if we choose a semi-implicit scheme for the time discretization, we have to solve the following problem :

Real part

$$\begin{cases} \phi_r^{n+1} - \alpha\Delta\phi_r^{n+1} = \mathcal{F}^{r,n} + \Delta t\phi_i^n & \text{in } \Omega_1, \\ -\Delta\phi_r^{n+1} = \mathcal{J}_r^{n+1} & \text{in } \Omega, \\ [\phi_r^{n+1}] = 0 \quad \text{and} \quad [\dfrac{\partial\phi_r^{n+1}}{\partial n}] = 0 & \text{in } \partial\Omega_1, \end{cases} \tag{3.42}$$

with $\mathcal{F}^{r,n} = \phi_r^n + \epsilon^2\Delta t(1-\theta)\Delta\phi_r^n + \Delta t C^{r,n}$ and $\alpha = \epsilon^2\theta\Delta t$.

Imaginary part

$$\begin{cases} \phi_i^{n+1} - \alpha\Delta\phi_i^{n+1} = \mathcal{F}_i^n - \Delta t\phi_r^n & \text{in } \Omega_1, \\ -\Delta\phi_i^{n+1} = \mathcal{J}_i^{n+1} & \text{in } \Omega, \\ [\phi_i^{n+1}] = 0 \quad \text{and} \quad [\dfrac{\partial\phi_i^{n+1}}{\partial n}] = 0 & \text{in } \partial\Omega_1, \end{cases} \tag{3.43}$$

with $\mathcal{F}^{i,n} = \phi_i^n + \epsilon^2\Delta t(1-\theta)\Delta\phi_i^n + \Delta t C^{i,n}$.

Here, α is a small parameter in both systems and θ is the Crank-Nicolson method parameter ($\theta \in [0,1]$).

Both systems "imaginary part" and "real part" are coupled, we use an explicit time discretization to express the coupling term. This method allow on the one hand to solve the coupling system and on the other hand to be easily parallelisable.

Spatial discrétization

Let :

- ϕ_r et ϕ_i, be the vectors solution of discretized problems corresponding to (3.42) and (3.43) systems such that :
  ϕ_r^1, ϕ_i^1 : ϕ_r respectively ϕ_i on the mesh nodes of Ω_1
  ϕ_r^0, ϕ_i^0 : ϕ_r respectively ϕ_i on the mesh nodes of Ω
- F_r and F_i be the respective values of $\mathcal{F}_r$ and $\mathcal{F}_i$ on the mesh nodes of Ω_1,
- J_r and J_i be the respective values of $\mathcal{J}_r$ and $\mathcal{J}_i$ on the mesh nodes of Ω,
- L_1 : associated operator to $-\Delta\phi$ on Ω_1,
- L : associated operator to $-\Delta\phi$ on Ω,
- I_1, I_0 : be the identity operators respectively on Ω_1 and Ω,
- D_1, D_0 : operators associated to $\dfrac{\partial}{\partial n}$ respectively on $\partial\Omega_1$ and $\partial\Omega$.

We have :

Real part:

$$\begin{cases} -\alpha L_1\phi_r^{1,n+1} + I_1\phi_r^{1,n+1} & = F_r^n + I_1\phi_i^{1,n} \\ L\phi_r^{0,n+1} & = J_r^{n+1} \\ [\phi_r^{n+1}] & = 0 \\ D_1\phi_r^{1,n+1} & = D_0\phi_r^{0,n+1}, \end{cases} \tag{3.44}$$

Imaginary part:

$$\begin{cases} -\alpha L_1 \phi_i^{1,n+1} + I_1 \phi_i^{1,n+1} & = F_i^n + I_1 \phi_r^{1,n+1} \\ L\phi_i^{0,n+1} & = J_i^{n+1} \\ [\phi_i^{n+1}] & = 0 \\ D_1 \phi_i^{1,n+1} & = D_0 \phi_i^{0,n+1}. \end{cases} \tag{3.45}$$

Resolution of the transfer problem

We solve the transfer problem between Ω_1 and Ω_0 with *Funaro-Quarteroni (F.Q) procedure*

If we write:

- $\phi_r^{1,n+1,p}$ (resp $\phi_r^{0,n+1,p}$) the vector of values of ϕ_r on the mesh nodes Ω_1 (resp Ω) at time iteration n+1, for the (F.Q) iteration p,
- $\phi_i^{1,n+1,p}$ (resp $\phi_i^{0,n+1,p}$) the vector of values of ϕ_i on the mesh nodes Ω_1 (resp Ω) at time iteration n+1, for the (F.Q) iteration p.

We write the algorithm in the following way:

while $(t_n < t_{max})$
p=0 (p: index of the (F.Q.) algorithm)
while (criterion to stop (F.Q) algorithm)

$$\begin{cases} -\alpha L_1 \phi_r^{1,n+1,p} + I_1 \phi_r^{1,n+1,p} & = F_r^n + I_1 \phi_i^{1,n,p-1} \\ D_1 \phi_r^{1,n+1,p} & = D_0 \phi_r^{0,n+1,p-1} \\ L\phi_r^{0,n+1,p} & = J_r^{n+1} \\ \phi_r^{0,n+1,p} & = \phi_r^{1,n+1,p} \\ -\alpha . L_1 \phi_i^{1,n+1,p} + I_1 \phi_i^{1,n+1,p} & = F_r^n + I_1 \phi_r^{1,n,p} \\ D_1 \phi_i^{1,n+1,p} & = D_0 \phi_i^{0,n+1,p-1} \\ L\phi_i^{0,n+1,p} & = J_i^{n+1} \\ \phi_i^{0,n+1,p} & = \phi_i^{1,n+1,p} \end{cases}$$

end while (F.Q)
$t_n = t_n + \Delta t$
end while (time itération)

Remarks

The coupled term between "real part" and "imaginary" part has been expressed with an explicit way in the time discretization method, we want to observe how the error propagates if we keep this term implicit.

For that, we only need to keep, in a one dimensional space, the idea of demonstration for the lemma 3. The results obtained will spread in a two dimensional space, as for the

theorem (7) , via a comparison lemma.

If we write (e_j^r) the error on ϕ_j^r, respectively (e_j^i) the error on ϕ_j^i, e_j^r and e_j^i verify the system:

$$\begin{cases} \phi_i - \alpha\dfrac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{h^2} + \Delta t\phi_i &= 0 \\ \psi_i - \alpha\dfrac{\psi_{i+1} - 2\psi_i + \psi_{i-1}}{h^2} - \Delta t\psi_i &= 0. \end{cases}$$

Let $c = \dfrac{\alpha}{\Delta t h^2} = \dfrac{\epsilon}{h^2} \ll 1$, we obtain the connecting equation

$$\psi_i = c(\phi_{i+1} + \phi_{i-1}) - (2c + \tfrac{1}{\Delta t}). \tag{3.46}$$

In so doing, carrying in the ψ equation

$$\mu\phi_{i+2} + \lambda\phi_{i+2} + \gamma\phi_i + \lambda\phi_{i-1} + \mu\phi_{i-2} = 0,$$

with

$$\begin{aligned} \mu &= c^2 = \frac{\epsilon^2}{h^4}, \\ \lambda &= -2c(2c + \frac{1}{\Delta t}), \\ \gamma &= (5c^2 + \frac{4c}{\Delta} + \frac{1}{\Delta^2} + 1). \end{aligned}$$

We can write

$$\phi_i = \sum_{j=1}^{4} \lambda_j r_j^i$$

where r_j are the four roots of the reciprocal equation

$$\mu(r^4 + 1) + \lambda(r^3 + r^2) + \gamma r^2 = 0.$$

The λ_j parameters are determined from the boundary conditions. If r is a root of the reciprocal equation, $\dfrac{1}{r}$ is also a root. We can write the equation in the following way

$$\mu(r + \frac{1}{r}) + \lambda(r + \frac{1}{r}) + \gamma = 0.$$

We put $X = r + \dfrac{1}{r}$, in this case we write the equation

$$\mu X^2 + \lambda X + \gamma = 0.$$

the both solutions are:

$$\begin{aligned} X_1 &= -(4 - \frac{2}{c\Delta t}) - \frac{2}{c}i, \\ X_2 &= -(4 - \frac{2}{c\Delta t}) + \frac{2}{c}i. \end{aligned}$$

Asymptotically, we can write

$$\begin{aligned} X_1 &\sim -\tfrac{1}{c} \\ X_2 &\sim \tfrac{1}{c}, \end{aligned}$$

and

$$\begin{aligned} r_1 &\sim -\frac{i}{c} \\ r_2 &\sim \frac{1}{c}. \end{aligned}$$

We have $e_i \sim \lambda r_1^i + \mu r_2^i$, using *the boundary conditions* to compute λ and μ.

Asymptotically, the roots of $(e^i_{j,k})$ and of $(e^r_{j,k})$ are the same. The boundary conditions which allow to compute the factors $(\lambda_j)_{j=1,\dots 4}$ are the same for both problems, the errors behavior is then identical. It is easy to show, by the same principle we used for the lemma 2, using the boundary conditions both sides of the surface conductor, and the coupled equations that the damping factor $\xi_{\leftarrow} \ll 1$.

3.4.3 Non-stationary case

From the simplified model formulation studied in the previous chapter and the used operators analysis, we made a feasibility study.

To resolve the model (2.28), we amount to the following problem for each iteration (2.29) :

$$\begin{cases} \omega - \alpha\Delta\omega = \mathcal{F} & \text{in} \quad \Omega_1, \\ \omega = 2 \times \dfrac{\partial^2\psi}{\partial r^2} & \text{on} \quad \partial\Omega, \\ -\Delta\psi = \omega & \text{in} \quad \Omega_1, \\ \psi = 0 & \text{on} \quad \partial\Omega_1. \end{cases} \tag{3.47}$$

We have a coupled problem where the operators are $Id - \alpha\Delta$ and Δ with Δ, the "Laplacian" operator and α, our problem small parameter.

- $Id - \alpha\Delta$: α is a small parameter, for that reason we have a boundary layer phenomena. The (F.Q.) procedure can be used, we don't need a domain decomposition method without recovering, therefore we can use the Schwarz method, fully adapted to this kind of singular perturbation problem ([28], [29],[30],[27],
- for the "Laplacian" operator, all classical methods can be used,
- an Arakawa's second order scheme ([2]) can be used for the "Jacobian" discretization which is a square vorticity conserving scheme and an energy conserving scheme.

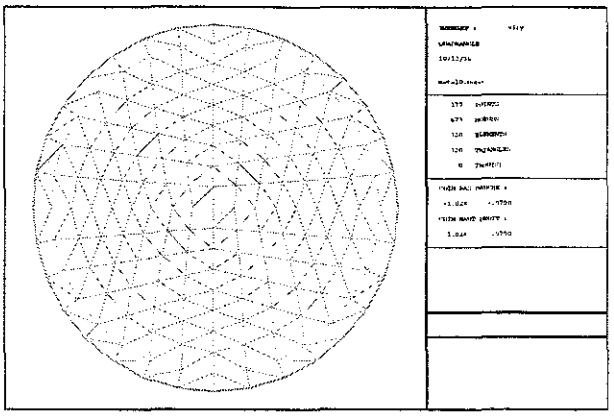


FIG. 3.5 – *mesh of the outer subdomain on the metal for N=10*

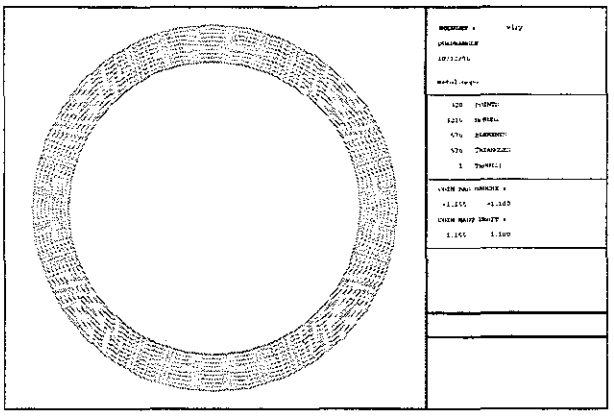


FIG. 3.6 – *mesh on the inner subdomain on the metal for N=10*

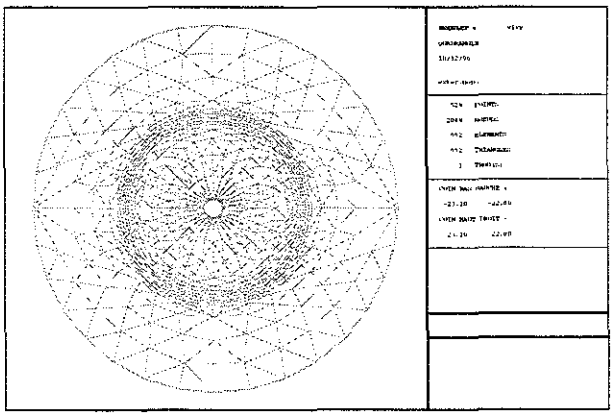


FIG. 3.7 – *mesh on the external domain for N=10*

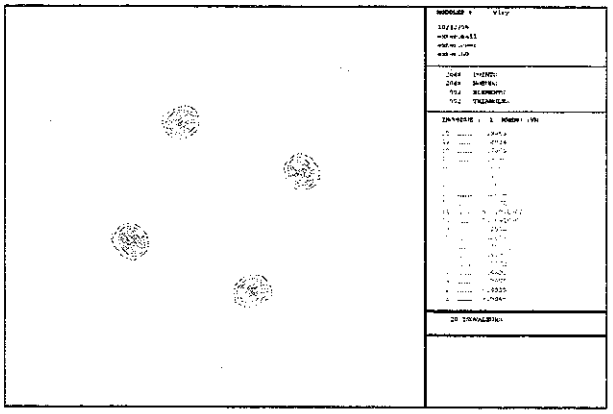


FIG. 3.8 – *isovalue of j on the external domain for N=10*

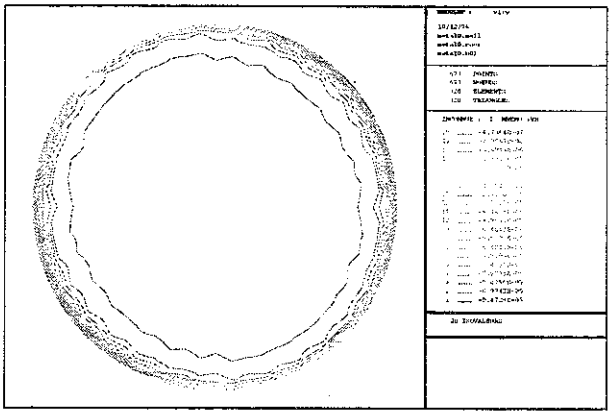


FIG. 3.9 – *isovalue on* Ω^1_{outer} *for N=10*

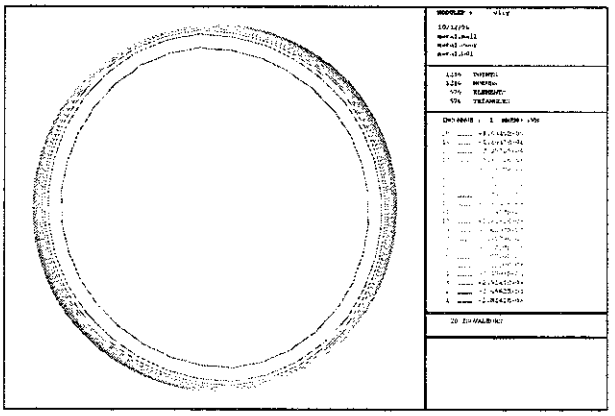


FIG. 3.10 – *isovalue on* Ω^1_{inner} *for N=10*

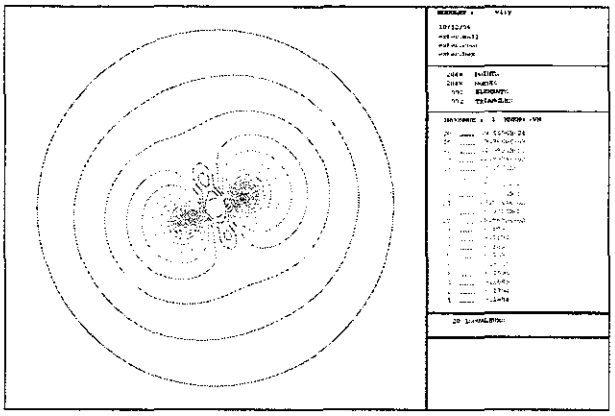


FIG. 3.11 – *isovalue on* Ω^2 *for N=10*

BIBLIOTHÈQUE UNIVERSITÉ NANCY 1 SCIENCES

Chapitre 4

Modèles et langages de programmation Application au développement d'un code PIC

4.1 Introduction

L'activité de programmation est abordée de façon diverse et controversée. Suivant l'importance du projet et la volonté d'extensibilité et de réutilisabilité, elle ne peut souvent pas se résoudre à un simple enchainement d'instructions ou d'appels de sous-programmes. Dans l'application décrite précédemment, l'utilisation d'un même algorithme sur différentes parties de l'application aurait dû permettre un passage du cas stationnaire au cas non stationnaire relativement rapide. Or l'utilisaton, entre autres, du langage Fortan77 et des techniques classiques de super tableau permettant la gestion de l'ensemble des structures de données dans la bibliothèque Modulef a rendu la réutilisation du code existant lourde et surtout peu fiable, toute modification se répercutant de façon difficilement contrôlable dans le reste du code.

Dans le premier paragraphe de ce chapitre, après avoir défini les critères d'un code "général" suivant l'utilisation qui en est faite, nous verrons comment la programmation orientée objet (P.O.O) peut répondre à ces critères et comment peut se faire le choix du ou des langages de programmation dans le domaine du calcul scientifique. Notre argumentation s'appuiera sur le développement d'un code PIC (particle-in-cell) de la physique des plasmas.

Dans les paragraphes suivants, nous appliquons les principes observés au développement d'un modèle objet de base utile au développement d'un code PIC. Après avoir décrit comment le passage du modèle physique à un modèle mathématique nous permet de définir un modèle objet de base adapté, nous décrivons l'implémentation d'un modèle discrétisé à partir de ce modèle objet de base (commun à tous les modèles discrétisés). Cette implémentation met en évidence les facilités d'extensibilité et la fiabilité de notre modèle.

Le développement de notre code s'est situé dans le cadre d'un projet de physique des plasmas avec E.Sonnendrucker[1] utilisant les méthodes particulaires pour effectuer la simulation numérique de l'évolution du plasma étudié. Un plasma de faible densité possède un trillion de particules par cm^3. Notre implémentation doit nous permettre de s'approcher autant que possible d'un système physique tout en tenant compte des contraintes des calculateurs (taille mémoire, CPU,...). Pour permettre de traiter ces grandes quantités de données avec des temps CPU raisonnables, notre code prévoit dans sa conception une implémentation parallèle rapide adaptée soit à une architecture mémoire distribuée, soit à une architecture mémoire partagée. Pour cela la conception du modèle objet est totalement adaptée à une méthode de résolution par décomposition de domaine ce qui permet une implémentation parallèle de type SPMD de façon assez naturelle.

4.2 Modèle et langage de programmation

4.2.1 Introduction

On définit les caractéristiques d'un code "général" suivant le type d'utilisation qu'il en est fait. On considère trois types d'utilisation qui correspondent :

- Soit à *une simple paramétrisation du code*, le code devra être simple à paramétrer et il devra permettre un contrôle efficace des erreurs dues à un mauvais paramétrage,
- soit à une adaptation du code existant à de nouveaux cas tests. Cette adaptation ne devra pas nécessiter de gros développements et ne devra pas entrainer des *effets de bords* indésirables dans tout le reste du code.
- soit à l'intégration facile de nouvelles méthodes de résolution, de discrétisation, de correction et d'interprétation des résultats...

De plus si ce code est destiné à être utilisé par diverses personnes sur des cas tests et des architectures différentes, on ne pourra négliger le critère de *portabilité.*

Définissons les caractéristiques d'un code "général" par la liste de critères suivante :

- La *Capacité* à s'adapter à la complexité des problèmes physiques traités.
- L'*exactitude* : aptitude de ce code à fournir les résultats voulus, dans des conditions normales d'utilisation.
- La *robustesse* : aptitude à bien réagir lorsqu'on s'écarte des conditions normales d'utilisation (mauvais paramétrage par exemple).
- L'*extensibilité* : facilité avec laquelle le programme pourra être adapté pour satisfaire une évolutions des spécifications, l'intégration de nouvelles méthodes de résolution ou de discrétisation.
- La *réutilisabilité* : possibilité d'utiliser certaines parties du code pour résoudre de nouveaux cas tests.

1. CNRS - IECN - Université Henri Poincaré, F-54506 Vandoeuvre les Nancy

- La *portabilité*: facilité avec laquelle on peut exploiter un même code dans différentes implémentations.
- L'*efficience*: aptitude à intégrer les concepts d'optimisation du temps CPU, de la taille mémoire utilisée, de l'utilisation de l'architecture de la mémoire...

Nous allons voir dans ce qui suit que c'est l'utilisation des concepts de base de la programmation orientée objet (P.O.O) qui semble être la meilleure réponse à ces exigences.

4.2.2 Programmation Orientée-Objet

Nous rappelons les principaux concepts de base de la P.O.O.

- **Objet**: La P.O.O est basée sur la notion d'objet, à savoir une association des données et d'algorithmes (méthodes) agissant sur ces données.
- **Encapsulation**: Cette association est plus qu'une simple juxtaposition. L'encapsulation des données signifie qu'il n'est pas souhaitable, et de ce fait rendu impossible d'agir directement sur les données d'un objet sans passer par l'intermédiaire de ses méthodes. Elles jouent ainsi le rôle d'interface suffisante et obligatoire.

  Le grand mérite de l'encapsulation est que, vu de l'extérieur, un objet se caractérise uniquement par les spécifications (noms, arguments et rôles) de ses méthodes. La manière dont sont réellement implantées les données est alors invisible, sa connaissance étant sans utilité à l'extérieure. On décrit souvent une telle situation en disant qu'elle réalise une "**abstraction des données**". On peut remarquer qu'en programmation structurée, une procédure pouvait également être caractérisée (de l'extérieur) par ses spécifications, mais que, faute d'encapsulation, l'abstraction des données n'était pas complétement réalisée.

  L'encapsulation des données présente un intérêt en matière de qualité de logiciel. Elle facilite la maintenance: une modification éventuelle de la structure de données interne d'un objet, tant qu'elle n'intervient pas sur les spécifications des méthodes, n'a d'incidence que sur l'objet lui même; les utilisateurs de l'objet ne seront pas concernés par la teneur de cette modification (ce qui n'est généralement pas le cas en programmation structurée). De la même manière, l'encapsulation des données facilite grandement la réutilisation d'un objet: l'utilisation qui est faite de l'objet ne doit alors pas intervenir sur son fonctionnement interne.
- **Classe**: Le concept de classe est à la base de l'abstraction des données. Ce dernier correspond à la généralisation de la notion de type que l'on rencontre dans les langages classiques. Une classe n'est rien d'autre que la description d'un modèle pour des objets ayant une structure de données commune et disposant des mêmes méthodes. Les objets apparaissent alors comme des variables d'un tel type classe, chacun disposant d'un exemplaire propre de la structure de donnees, et chacun pouvant voir appliquer les méthodes sur ses données propres.

- **Héritage**: Le concept d'héritage est la notion fondamentale en P.O.O. Il permet de définir une nouvelle classe à partir d'une classe existante, à laquelle on ajoute de nouvelles données et/ou de nouvelles méthodes. La conception d'une nouvelle classe, qui "hérite" les propriétés et les aptitudes d'une ancienne, peut ainsi s'appuyer sur des réalisations antérieures parfaitement au point et les "spécialiser" à volonté. Le concept d'héritage est à la base des possibilités de réutilisation de produits existants, et ceci d'autant plus qu'il peut être itéré autant de fois que nécessaire. Cette technique va donc permettre de développer de nouveaux outils en se fondant sur un certains acquis.
- **Polymorphisme**: Le polymorphisme est un concept de la théorie des types dans laquelle un nom peut désigner des objets de beaucoup de classes différentes qui sont reliées, par héritage, par un certaine superclasse commune. Ainsi, tout objet désigné par ce nom est capable de répondre de différentes manières à un certain ensemble commun d'opérations.

La P.O.O est une technique permettant la gestion de la complexité en associant chaque objet à une classe placée dans une hiérarchie de classes *de plus en plus spécialisées*. Un objet se souvient de sa propre classe même si il peut être pris pour un objet d'une classe supérieure dans la hiérarchie.

Si la programmation objet est la possibilité de définir des types abstraits ainsi que la possibilité de réutiliser des codes existants, *elle est avant tout la possibilité de modéliser la complexité des problèmes.* La complexité d'un problème pouvant être réduite en hiérarchisant les différentes complexités. La P.O.O engendre *une hiérarchie de type d'objets* beaucoup plus puissante que *la hiérarchie de procédures* (programmation structurée) ou *la hiérarchie d'objets imbriqués* (programmation usant de type abstrait).

4.2.3 Pourquoi la P.O.O?

Historiquement, les codes de physique des plasmas ont été programmés en FORTRAN et plus récemment en C. Ces codes ont été écrits pour résoudre des problèmes spécifiques et nécessitent un spécialiste pour leur utilisation. Plus récemment, des codes plus généraux adaptés à des familles de problèmes ont été développés (Magic [32], Isis [12], PDx). Ils utilisent une programmation structurée qui permet d'en améliorer l'exactitude et la robustesse mais on observe encore une certaine rigidité en ce qui concerne le choix des paramètres et des algorithmes.

En effet, en programmation structurée, on a une faible encapsulation des données du fait du couplage de fonctions indépendantes qui dépendent de l'implémentation d'autres mêmes fonctions. On pourra observer des effets de bord indésirables lors de la mise à jour du code. La P.O.O fournit une solution à ce problème par l'encapsulation de données.

L'intégration d'un nouveau modèle ou d'un nouvel algorithme dans un code structuré reste délicate parce qu'elle va engendrer la reécriture ou la mise à jour d'un certain nombre

de fonctions. Il sera souvent nécessaire de changer le paramétrage de fonctions existantes, de modifier le code de fonctions appelantes...Ces types de modifications, aussi simples soient-elles, peuvent provoquer des erreurs qui pourront se propager de manière difficilement contrôlable dans le reste du code. Un temps de débogage non négligeable sera donc nécessaire à l'adaptation de tout nouveau modèle, comme à l'ajout d'un nouvel algorithme.

Il faut remarquer qu'en même temps que le nombre des algorithmes permettant de résoudre des modèles complexes de la physique des plasmas augmente, les interactions entre ces algorithmes deviennent de plus en plus complexes à analyser. La programmation structurée permet de résoudre une partie de la complexité de notre modèle physique par une hiérarchisation de procédures mais ne fournit pas, on l'a vu, de solution simple à ce type de complexité qui sera prise en compte, nous allons le montrer, de manière naturelle dans un modèle orienté objet grâce aux concepts de classe, d'héritage et tout particulièrement de polymorphisme.

En P.O.O, on définit des objets qui représentent un concept ou des entités proches du domaine d'application et on intégre ces abstractions dans le code de calcul. Chacun de ces objets est composé de données et de méthodes qui interviennent de manière interne ou externe sur les données. Par le procédé d'encapsulation qui consiste à cacher les données et les méthodes internes à l'utilisateur, l'auteur de l'objet peut modifier la représentation interne de l'objet sans perturber le reste de l'application qui utilise cet objet. On peut donc créer une interface fixe entre les objets et atteindre les critères de réutilisabilité et d'extensibilité, bien utiles dans la construction par étapes ou en équipe d'une application, puisque les composants sont séparés.

Le langage C, comme la plupart des langages évolués, utilise la surdéfinition d'un certain nombre d'opérateurs. Un des grands atouts de la programmation objets est de permettre la surdéfinition de la plupart des opérateurs lorsqu'ils sont associés à la notion de classe. Cette possibilité est basée sur la surdéfinition de fonctions et le choix de la fonction utilisée est déterminé par le type des arguments.

Mais ce qui distingue la P.O.O des autres modèles de programmation, c'est avant tout qu'elle fournit la possibilité d'utiliser le concept d'*héritage*. Cela signifie qu'un objet peut hériter les données et les méthodes d'un objet "parent" et qu'on peut lui définir des propriétés spécifiques. *L'héritage peut donc être un moyen d'adapter à un besoin spécifique un objet défini de manière plus générale.* L'héritage fournit un potentiel évident de réutilisation de code. Il est également utile pour établir les liens au sein d'une hiérarchie de modules qui composent une application. Une hiérarchie adaptée peut-être une aide efficace dans la gestion de la complexité d'une application.
Le *polymorphisme* offre la possibilité pour une fonction d'avoir un comportement différent suivant le type d'objet sur lequel on l'invoque. Il est donc possible, pour une classe héritante de redéfinir une méthode de la classe "parent", ce qui correspond à une redéfinition du comportement de l'objet héritant sans perturber le reste du code. En effet, à l'exécution, chaque objet connait implicitement son type et invoque dynamiquement, soit la méthode associée à son type, si elle a été implémentée, soit la méthode de son plus proche

parent. Grâce à l'utilisation du polymorphisme, l'objet ne devra pas nécessairement contenir tous ses comportements possibles, implémentés par des blocs IF..ELSE ou SWITCH, il suffira de définir une nouvelle classe héritée pour chaque nouveau comportement. Le polymorphisme est une solution simple à la gestion de la complexité des interactions entre tous les algorithmes.

4.2.4 Choix du langage

Dans le domaine scientifique, le choix des langages de programmation est controversé. Fortran était depuis de nombreuses années le standard et il garde encore l'avantage, bien que le développement de bibliothèques en C et C++, de plus en plus important, laisse encore aujourd'hui subsister le retard du langage Fortran. C offre des possibilités de gestion de mémoire et de structure de données, C++ permet l'utilisation de l'intégralité des concepts de la P.O.O. On peut toutefois préciser que la nouvelle norme fortran (Fortran90/95 [37]) a comblé, entre autre, les lacunes de gestion de mémoire et de capacité d'abstraction.

Le langage C++ se situe entre le point de vue de la P.O.O pure comme Simula, Smalltalk et Eiffel, où tout programme écrit dans l'un de ces langages utilise obligatoirement une technique de P.O.O, et celui de la programmation structurée comme C, Fortran ou Pascal. La solution apportée par B. Stroustrup a le mérite de préserver l'existant (compatibilité avec les programmes écrits en C). Il a en effet été conçu en ajoutant à un langage classique (C) les outils permettant de mettre en oeuvre tous les principes de la P.O.O mais il n'impose nullement l'application stricte des principes de P.O.O. En effet, rien n'empêche (sauf notre bon sens) de faire cohabiter des objets réalisant une parfaite encapsulation de leurs données, avec des fonctions classiques réalisant des effets de bords sur des variables globales...Cette conception, très dangereuse dans la majorité des cas peut permettre toutefois la récupération d'algorithmes performants existants.

Les capacités de C++ dans le domaine de la programmation objet sont clairement établies ([13],[14],[69]), tandis que l'utilisation de certains concepts objets avec la nouvelle norme Fortran (Fortran90/95) est une nouveauté ([20]).

Lorsqu'il s'agit de choisir un langage, les problèmes qui se posent sont :

- Qu'en est-il des performances de C++ dans le domaine des applications scientifiques, de la facilité et des performances du traitement des tableaux?
- Jusqu'ou va la nouvelle norme Fortran dans l'intégration des concepts de base de la manipulation des objets?

D'une étude comparative entre C++ et Fortran90 dans le domaine de la programmation scientifique orientée objets ([15]) et d'un certain nombre d'échanges sur des listes de diffusion (comp-fortran-90, oon-digest, blitz-support), il ressort :

P.O.O : que contrairement à C++, Fortran90/95 n'intégre pas tous les concepts de la P.O.O. En effet, le concept d'objet est exprimé par deux éléments de la syntaxe : la

définition d'un *type* (structure de données) et la définition d'un *module* contenant la définition de ce type et les méthodes qui lui sont associées (surcharge d'opérateur, interface générique,méthodes privées ou publiques ...). L'objet hérité a seulement la particularité d'avoir pour composante l'objet "parent", il n'est pas un objet de type "parent" à part entière. Cette approche, si elle permet l'encapsulation et la programmation générique ne permet pas de construire une hiérarchie d'objets.
D'autre part, l'absence de la technique de "pointer casting" élimine toute possibilité d'utiliser le concept de *polymorphisme.* Cela implique que les critères de réutilisabilité, d'extensibilité et de hiérarchisation des complexités ne peuvent pas toujours être obtenus avec Fortran90 et sont beaucoup plus faciles à atteindre avec C++.

Performance Aujourd'hui, le compilateur ne se contente plus de traduire un code source en langage évolué vers un code en langage machine, il se charge également de l'optimisation sur la machine cible. Et on peut affirmer que la puissance d'optimisation du compilateur prend une part aussi importante que la nature du processeur ou de la mémoire hiérarchique. L'arrivée des processeurs RISC et des processeurs super-scalaires vont acroître la dépendance entre compilateur et performance. Le compilateur est un composant fondamental de l'optimisation de code et c'est encore le langage Fortran qu'il sera plus efficace d'utiliser pour obtenir de bonnes performances sur une grande variété d'architectures. Pas seulement, parce que les constructeurs continuent à fournir un gros effort de développement et de maintenance en Fortran mais aussi pour l'utilisation, entre autres, des pointeurs dans les autres langages, qui sont, pour la plupart des compilateurs, dans certaines circonstances des inhibiteurs de performance; faute d'information suffisante, le compilateur adopte une attitude conservative et génère un code moins optimisé ([22]).

Cependant des progrès ont été faits et des tests ont été effectués sur une variété de plate-formes (HP, SGI, SUN, ...), qui ont montré que l'utilisation de langages de procédures comme F77, C à la place de langage intégrant certains concepts de la P.O.O comme C++ et Fortran90 n'apportait qu'un faible gain de performance ([15]). Mais ces tests ne faisaient pas intervenir les concepts de la P.O.O qui sont la caractéristiques de langages comme C++ et pour certains de Fortran90.
Actuellement, il est admit, que c'est l'utilisation de ces concepts objets, très utiles au niveau du traitement de la complexité, de l'extensibilité et de la réutilisabilité d'une application, qui apporte un overhead et une perte de performance qu'on ne peut négliger ([61]). En effet :

- L'utilisation du concept de polymorphisme, synonyme en C++ d'utilisation de pointeurs, or l'utilisation des pointeurs et des phénomènes d'aliasing inhibent l'optimiseur de nombreux compilateurs C/C++, tout particulièrement le "Loop Nest Optimizer" principalement chargé de la bonne utilisation du/des caches. Il en sera de même de l'utilisation des tableaux multidimensionnels (absence d'objet tableau dans le langage C++)
- La plupart des compilateurs modernes favorisent l'allocation des variables scalaires dans les registres, plus rapides d'accès, et ne permettent pas l'allocation de variables non-scalaires dans ces registres. Ce qui explique la médiocrité des

performances de ces compilateurs C sur des programmes C++, essentiellement basés sur des variables objets, tout particulièrement dans le cas de "petits objets". On appelle "petits objets", les gros volumes de données composés de nombreux "petits objets" (tableau de nombres complexes) ou les gros volumes de données accédés par des "petits objets" (descripteurs de tableau et itérateurs).
- L'absence de connaissance sémantique d'un type abstrait utilisateur, fait que devant un objet de ce type, le compilateur voit un pointeur et des boucles et ne peut effectuer les optimisations classiques effectuées sur les types du langage.
- En Fortran90/95, les tableaux, objets très utilisés dans le domaine scientifique, sont définis dans le langage; l'optimisation de leur traitement prise en charge par le compilateur, est totalement adaptée à l'architecture (massivement paralèlle, simple processeur, multi-processeurs). D'autre part, la syntaxe d'utilisation de ces tableaux (matrices ou vecteurs) est très simple, voisine de celle utilisée dans le modèle mathématique qui définit les algorithmes.

Sans le développement de bibliothèques spécifiques, les tableaux sont un problème en C ou C++, pour les performances des applications scientifiques, grandes consommatrices de ce type d'objets.

Si on veut préserver les performances d'un code C++, les caractéristiques du langage qui doivent être utilisées avec précaution sont les "petits objets", les fonctions virtuelles et les exceptions. L'utilisation des "petits objets" fait chuter considérablement les performances ([61], [68]) et contrairement à une idée bien répandue, c'est l'utilisation intensive de ces "petits objets", et non pas celle des fonctions virtuelles, outils d'implémentation du polymorphisme, qui est la principale cause de la dégradation des performances.

De nombreux compilateurs ne sont pas encore capables de gérer le niveau d'abstraction d'un programme orienté objets et lorsqu'on veut obtenir des performances, on a trois catégories de solutions.

- Ecrire les parties critiques en minimisant le niveau d'abstraction,
- Ecrire les parties critiques dans un langage de bas niveau d'abstraction comme Fortran ou C. Cette technique génère un overhead non négligeable, elle sera donc efficace sur des gros grains de calcul.
- Ecrire une implémentation plus judicieuse de l'abstraction. Des résultats ont été obtenus sans amélioration de l'optimisation des compilateurs ou extension du langage, mais uniquement grâce à une utilisation judicieuse des "template" par la technique des "*expressions template*" ([72]).

Déterminer le moyen le plus efficace d'utiliser la P.O.O, est une préoccupation importante de la recherche actuellement([38]) et un effort considérable a été fourni dans la communauté C++ Orienté-Objets.

- Amélioration des compilateurs C++ : le *compilateurs KAI C++* ([39]) développé par Arch.F Robison et ses collaborateurs résout une grande partie des pro-

blèmes majeurs de performances ([62]). Les solutions aux problèmes qui subsistent sont bien connues et pourraient être résolues mais le marché du calcul scientifique ne peut supporter économiquement leur développement. D'autre part, il est peu probable, qu'il existe une solution générale aux problèmes d'optimisation, chaque applications ayant ses propres particularités.

- Une autre axe, riche en développement actuellement, est de déplacer le haut-niveau d'optimisation hors des compilateurs vers des bibliothèques que l'on appelle **Bibliothèques Actives** ([78]). Une bibliothèque active définit ses propres abstractions, effectue une implémentation optimisée de leurs traitements grâce à la programmation générique ([55]), à l'optimisation générative ([78]) et à des techniques judicieuses utilisant l'outil de "template" figurant dans C++ comme les "expressions template" ([72]), l'évaluation partielle, ([77]) et les "metaprogram template" ([73]). Les "template" peuvent être utilisés pour effectuer des calculs à la compilation ([71]). Le but de ces bibliothèques est de combiner, la simplicité de l'utilisation, la performance du code et l'adaptatbilité, critères propres aux bibiothèques de haut niveau d'abstraction tout en fournissant une solution aux problèmes de performance inhérents à l'abstraction.

Dans le domaine du calcul scientifique, les abstractions qui sont couramment utilisées, sont *les tableaux, les vecteurs, les matrices et les tenseurs.* Il existe des développement importants de bibliothèques de type actif définissant ce type d'objets :

STL : Bibliothèque C++ ([54]) qui fournit des classes conteneurs, les itérateurs et les algorithmes génériques associés . Les conteneurs séquentiels principaux sont les classes "vector", généralisant la notion de tableau, la classe "list" celle de liste doublement chainée.

Blitz++ : C'est une bibliothèque qui a pour but de construire un environnement de base pour le calcul scientifique apportant à la fois abstraction et performance ([76], [75], [74]).

- elle fournit des objets génériques de type, vecteur, tableau et matrice, dans une syntaxe simple similaire à celle de Fortran90/95,
- sur les objets complétement implémentés, elle obtient des performance compétitives avec Fortran par l'utilisation des techniques de template,
- elle rajoute des caractéristiques que Fortran90/95 n'a pas telles que la transposition, la notation de tenseur, les réductions partielles,...

Cependant :

- Alors que l'implémentation des objets de type tableau et des expressions algébriques sur ces tableaux est simple syntaxiquement et déja performante ([75]), les matrices et les générateurs aléatoires ne sont que partiellement implémentés et l'utilisation immédiate d'un objet de type matrice ou tableau comme paramètre d'une routine de type LAPACK ne semble pas avoir été prévu.

- Blitz a été développé et testé sur AIX(IBM) avec le compilateur KAI (compilateur commercial), il est théoriquement stable sur CRAY, SGI, HP, SUN, Linux avec ce même compilateur.
- Blitz a utilisé pour son implémentation des caractéristiques de C++ qui ne sont pas encore supportées par tous les compilateurs (Intel C++, CC-SGI,..).

En conclusion, lorsque le but recherché par le développement d'un tel produit sera atteint, Blitz++ sera un outil intéressant pour allier performance, abstraction et simplicité dans le domaine du calcul scientifique. On peut même penser qu'étant donné qu'il implémente ses propres abstractions, on aura à terme une plus grande flexibilité quant au type des objets tableaux et du traitement sur ces objets. Mais des développements semblent encore nécessaires, tout particulièrement dans le domaine des objets matrices et de la possibilité d'utiliser simplement ces objets dans des procédures numériques, ainsi que pour la fiabilité et la portabilité de la bibliothèque.

POOMA II est une bibliothèque C++ qui utilise les mêmes techniques que Blitz. L'utilisateur écrit des expressions simples comme "A=B+C" qui déclenche la génération de routines "data-parallel" utilisant les threads ou le "message-passing" ([40]).

MTL (Matrix Template Library) [46] est une bibliothèque performante d'algèbre linéaire utilisant la programmation générique. Elle fournit des objets matrices de type dense, creuse, bande, symmetrique et triangulaire et les algorithmes numériques de bases sur ces objets. Une action combinée de l'optimisation des compilateurs C++ et de la techniques de "meta template" permet une optimisation des boucles. ([65]).

SL++ ([21]) : bibliothèque scientifique qui tente de répondre aux critères de performance, facilité d'utilisation et recouvrant un certain nombre de domaines du calcul numérique (algèbre, fft, quaternions, visualisation,...).

FFTw (the Fastest Fourier Transform in the West) ([1]),

Petsc ([4],[5]) : Librairie orientée objets de résolution d'équations aux dérivées partielles.

...

4.3 Du modèle physique vers un modèle objet adapté

4.3.1 Introduction

Dans ce paragraphe, nous décrivons comment le passage d'un modèle physique à un modèle mathématique puis finalement à un modèle discrétisé nous permet de définir un modèle objet adapté. Cette étude s'est faite dans le cadre du développement d'un code particulaire (PIC) de la physique des plasmas. L'expérimentation, outre le fait qu'elle est souvent très onéreuse, se heurte à des difficultés physiques pour effectuer certaines observations d'une part, d'autre part elle ne permet pas toujours une variation rapide

des paramètres caractéristiques ou un contrôle complet des influences extérieures sur le problème étudié. Certaines hypothèse simplificatrices ainsi que les données extraites de l'observation des expérimentateurs vont permettre de mettre en place un modèle mathématique. La simulation est, comme dans beaucoup d'autres domaines de la physique, une aide précieuse pour la conception et l'optimisation de nombreux dispositifs dans lesquels le principal phénomène physique mis en jeu est l'interaction d'un champ électromagnétique et d'un ensemble de particules chargées(ions et/ou électrons) : tubes électroniques, réacteurs de fusion nucléaire, diodes, etc...

Le modèle mathématique adapté à de tels dispositifs consiste en un couplage entre les équations de Maxwell déterminant les champs électromagnétiques auto-consistants engendrés par le plasma et l'équation de Vlasov décrivant l'évolution d'une espèce de particule, lorsque les collisions sont négligées. La simulation numérique de ce système peut être effectuée par un couplage entre une méthode de différences finies [11], élément finis [3] ou volumes finis pour la discrétisation des champs électriques et magnétiques et une méthode particulaire pour les particules chargées []. Dans le cadre des plasmas, la résolution numérique des équations de Vlasov décrivant le déplacement des particules est souvent réalisée par des méthodes PIC (Particle in Cell) qui consistent à modéliser l'évolution en temps de ces équations par un nombre fini de particules discrètes, chacune représentant un certain nombre de particules réelles chargées, leur trajectoire, solution de l'équation du mouvement, suivant les caractéristiques de l'équation de Vlasov. L'intéraction entre les champs (auto-consistants) et les particules se fait par **interpolation** par l'intermédiaire d'un maillage sur lequel on résout les équations de Maxwell. Interpolation de la densité de courant et de la densité de charge, engendrées par les particules, sur les noeuds du maillage pour calculer les termes sources des équations de Maxwell et l'interpolation des champs sur la position des particules pour calculer leur vitesse de déplacement à l'aide de l'équation du mouvement. Si on veut se rapprocher de façon précise du modèle physique, un grand nombre de particules est nécessaire et la résolution numérique du modèle complet peut se révéler coûteuse en temps de calcul et en place mémoire.

Le domaine d'observation peut-être divisé en plusieurs régions ayant des propriétés physiques différentes. Dans le modèle mathématique global, chaque région sera modélisée par des hypothèses et des équations qui lui sont propres. Enfin, les différentes régions interagissent entre elles, ainsi qu'avec le monde physique extérieur par l'intermédiaire des conditions frontières sur les champs et sur les particules. Le grand nombre de particules qu'il est nécessaire de considérer est une autre motivation de la décomposition en sous-régions du domaine d'observation. Dans ce cas également, les régions obtenues interagiront entre elles par l'intermédiaire des conditions frontières.

A partir du modèle mathématique, on peut chercher soit la solution analytique, lorsque le modèle est suffisamment simple, soit une solution numérique. Un grand nombre d'hypothèses simplificatrices doivent être faites pour résoudre le système analytiquement, aussi lorsque l'on veut se rapprocher du problème physique, on est amené dans la plupart des cas à déterminer une solution numérique et donc à définir un modèle discrétisé. Le modèle discrétisé est une approximation du modèle mathématique, il est totalement indépendant

de l'implémentation. Dans le cas d'un modèle de P.O.O, le choix des objets du modèle de base découle directement de l'analyse du modèle mathématique

Dans le cas d'une simulation utilisant une méthode particulaire, on distingue :
- les champs électriques et magnétiques connus sur les noeuds d'un maillage (discrétisation du domaine d'observation), solutions des équations de Maxwell,
- des groupes de macro-particules de types différents. Chaque macro-particule étant connues par sa position, sa vitesse, son poids (nombre de particules représentées par la macro-particule) et les caractéristiques de son groupe qui sont essentiellement la masse et la charge. Dans le cas d'une méthode particulaire, les équations de Vlasov sont remplacées par les équations du mouvement qui calculent la vitesse de chaque particule à partir de la valeur des champs sur cette particule.
- des frontières sur lesquelles seront définies les conditions limites des équations des champs et des particules. C'est aussi, au niveau des frontières que l'on s'appliquera à résoudre la parallélisation par décomposition de domaine.

Pour permettre le couplage entre le calcul des champs et le calcul du déplacement des particules, une interface est nécessaire entre les objets définissant les champs, les groupes de particules et les frontières. Elle devra permettre l'interpolation des champs sur la position des particules pour résoudre les équations du mouvement et l'interpolation des densités de charge et de courant sur les points du maillage pour permettre la résolution des équations de Maxwell.

4.3.2 Modèle mathématique

On se place donc dans le cas général d'un problème de évolution d'un plasma. Ce plasma peut souvent être considéré comme un gaz non collisionnel de particules chargées. Il est parfois plongé dans un champ magnétique B, statique ou lentement variable. Ce champ est produit par des sources extérieures au plasma ou engendré par les courants électriques associés au mouvement d'ensemble des particules du plasma. Il peut également être soumis localement à un champ macroscopique E dont l'origine peut être extétieure ou intérieure. Pour analyser le comportement de ce plasma, il est nécessaire de décrire le mouvement des particules chargées dans le champ (E,B). La simulation numérique de la circulation d'un tel plasma peut être obtenue par une méthode PIC (Particle-in-cell). Dans ce paragraphe, nous décrivons le modèle mathématique associé à cette méthode ainsi que le modèle objet de base associé. Puis dans le paragraphe suivant, nous décrirons le modèle objet associé à un modèle discrétisé à partir du modèle objet de base.

- Les champs sont calculés à partir des équations de Faraday et d'Ampère (unités mks-SI) :

$$\begin{cases} \dfrac{\partial B}{\partial t} &= -rotE, \\ \dfrac{\partial E}{\partial t} &= c^2 rotB - \frac{1}{\epsilon_0} J, \end{cases} \tag{4.1}$$

avec

$$divB = 0,$$

et l'équation de Poisson :

$$divE = \frac{\rho}{\epsilon_0},$$

où B est le champ magnétique, E le champ électrique, J la densité de courant et ρ la densité de charge. Les constantes c et ϵ_0 sont respectivement la vitesse de la lumière et la permittivité diélectrique du domaine.

Les densités de charge et de courant vérifient l'équation de conservation de la charge :

$$divJ + \frac{\partial \rho}{\partial t} = 0.$$

Lorsque l'équation de conservation de la charge est vérifiée, l'équation de Poisson est consistante avec (4.1).

- Pour analyser le comportement du plasma, il est nécessaire de décrire le mouvement des particules chargées dans le champ (E,B). Cela consiste à trouver les trajectoires qui correspondent à l'équation du mouvement d'une particule :

$$\begin{cases} m\dfrac{dv}{dt} &= q(E + v \wedge B), \\ \dfrac{dx}{dt} &= v \end{cases} \tag{4.2}$$

où x est la position d'une particule, v sa vitesse, m et q respectivement sa masse et sa charge.

Nous décrirons l'algorithme de résolution gobale, indépendament du type de discrétisation du domaine d'observation, du type de résolution des équations et du type de conditions limites et initiales. Le but étant l'implémentation *d'un modèle objet de base* à partir duquel se fera l'implémentation d'un nouveau cas test, d'une nouvelle méthode de résolution ou de nouvelles conditions limites.

Les champs électromagnétiques sont calculés à partir de (4.1) sur les noeuds d'un maillage du domaine d'observation. La densité de courant et la densité de charge, termes sources de (4.1) sont obtenues par interpolation à partir de la connaissance de la position et de la vitesse de toutes les particules. Les particules se déplacent sous l'effet des nouveaux champs électromagnétiques et des champs extérieurs éventuels. Après interpolation de ces derniers sur les particules et résolution de (4.2), on obtient leur nouvelle vitesse et leur nouvelle position.
Cette procédure se répéte sur un certain nombre de pas de temps. On note :

- **i** : l'indice d'une particule,
  x_i sa position et v_i sa vitesse,
- **j** : l'indice d'un noeud du maillage,
  $(E,B)_j$ les valeurs des champs électromagnétiques sur ce noeud.

On a l'algorithme :

- Initialisation :
  1. t=0, Δt ,
  2. Initialiation d'un maillage,
  3. Initialisation des paramètres physiques et numériques,
  4. Génération des particules dans l'espace des phases (x,v),
  5. Initialisation de E,B avec $divB = 0$ et $divE = \frac{\rho}{\epsilon_0}$,
  6. Autres initialisations spécifiques au cas test.
- Itérations en temps

  Le cycle d'un pas de temps se décompose de la façon suivante :
  1. Calcul des forces des équations des champs : Interpolation des positions et des vitesses des particules sur les noeuds du maillage
  $$(x,v)_i \rightarrow (\rho,J)_j.$$
  2. Intégration des équations des champs (4.1)
  $$(\rho,J)_j \rightarrow (E,B)_j.$$
  3. Calcul des forces sur les particules : Interpolation des champs sur les particules.
  $$(E,B)_j \rightarrow (E,B)_i.$$
  4. Intégration des équations du mouvement (4.2)
  $$(E,B)_i \rightarrow (x,v)_i.$$
- Diagnostiques qui dépendent des motivations de la simulation :
  1. impression des densités de charges ou de courant,
  2. valeurs des champs,
  3. analyse de l'évolution en temps de certains paramètres comme l'énergie, la position des particules...
  4. distribution des vitesses,
  5. ....

Remarques

1. L'algorithme décrit, est volontairement simplificateur, dans le sens où il ne considère que la résolution des paramètres du problème (E,B,x,v) sans tenir compte du traitement concernant le transfert des particules vers l'extérieur ou l'intérieur du domaine ou vers une zone voisine dans le cas d'une décomposition de domaine.
   De façon générale, pour tenir compte de ces transferts, à chaque itération en temps, l'algorithme doit :
   - commencer par un traitement de réception des particules entrant dans le domaine, avant le calcul des forces des équations des champs. C'est à ce moment que l'on traitera l'émission de particules par des éléments frontières.

- terminer, après le calcul du déplacement des particules, par une localisation dans le domaine de calcul courant de ces particules et le stockage , par type, des particules sortantes. C'est à ce moment que l'on traitera le cas de la réflexion.

2. Dans le cas d'une décomposition de domaine, on fera une implémentation parallèle en utilisant un modèle SPMD ou chaque zone sera traitée par un processeur avec le même algorithme. Les zones interagiront entre elles par l'intermédiaire des conditions frontières sur les champs et sur les particules. *C'est donc au niveau de l'implémentation de ces conditions frontières que se localisera principalement la parallélisation.*

4.3.3 Modèle objet adapté

Schéma général de l'application

Le schéma général de l'application, point commun de l'implémentation des cas tests, est le suivant (voir Fig:4.1) :

- L'espace étudié est divisé en zones indépendantes afin de permettre, les calculs sur des zones ayant des caractéristiques physiques différentes et une implémentation parallèle SPMD.
- L'algorithme PIC est géré sur chaque zone par un objet instantié à partir de la classe permanente "**zone**".
- Cette classe "**zone**" est rendue indépendante de l'implémentation du calcul des champs électromagnétiques et du calcul du déplacement des particules par l'usage de deux classes spécifiques gérant ces données. Elle est donc seulement gestionnaire des échanges entre ces classes spécifiques. Elle permet également la gestion des particules entrantes et sortantes à partir des classes "**frontiereParticules**".
- La gestion du calcul des champs est assurée par la classe "**champs_discrets**", classe conteneur ne disposant que *de méthodes virtuelles* à écrire dans des classes dérivées selon la modélisation choisie pour la résolution des équations de Maxwell et des différentes interpolations. Un objet de ce type sera instantié pour chaque zone.
- Le traitement des conditions limites sur les champs dans la résolution des équations de Maxwell est assurée par la classe "**frontiereChamps**". Cette classe est une classe conteneur dont *une méthode virtuelle* permet de modéliser les conditions limites sur les champs du cas test étudié. Elle sera appelée par la méthode de résolution des champs de la classe "champs_discrets" .
- La gestion des particules se fait par groupe de particules identiques partageant les mêmes particularités physiques. Elle est assurée par la classe permanente "**groupeParticules**", dont *une méthode virtuelle* permet de modéliser le déplacement des particules du groupe, par la résolution des équations du mouvement. Un nombre quelconque d'objets de ce type sera instantié sur chaque zone, en fonction du nombre de types de particules physiques mis en cause.
- Pour chaque zone et chaque itération en temps, il est nécessaire de considérer :
  - Les particules sortant de la zone après le calcul de leur déplacement par résolution des équations de mouvement. On envisage deux cas, le cas des particules

sortant du domaine d'observation et le cas des particules en transfert vers une zone voisine.

- les particules entrantes issues d'une zone voisine dans le cas d'une décomposition de domaine ou d'un élément frontière emmetteur ou réfléchissant.

Ces échanges entre zones sont modélisés par la classe conteneur "**frontiereParticules**", dont les *méthodes virtuelles* permettent de construire des processus d'échange de particules avec les zones voisines ou avec l'extérieur du domaine, et ce indépendemment du mode de calcul de chaque zone (séquentiel ou parallèle). Un nombre quelconque d'objets de ce type sera instantié sur chaque zone, en fonction de la géométrie et des conditions limites. Comme pour le calcul des champs et celui du déplacement des particules, la classe "zone" a un rôle de "chef d'orchestre" dans le problème du mouvement des particules sur les différentes zones du domaine d'observation.

- Naturellement, il va de soi que l'instantiation des éléments frontières est directement liée à la géométrie des zones, de même que celle du modèle des champs. Il devra donc exister des liens géométriques directs entre les classes dérivées de "champs_discrets" et "frontiereParticules" et "frontiereChamps", mais ces liens, totalement dépendants des modèles, ne peuvent être définis ici.

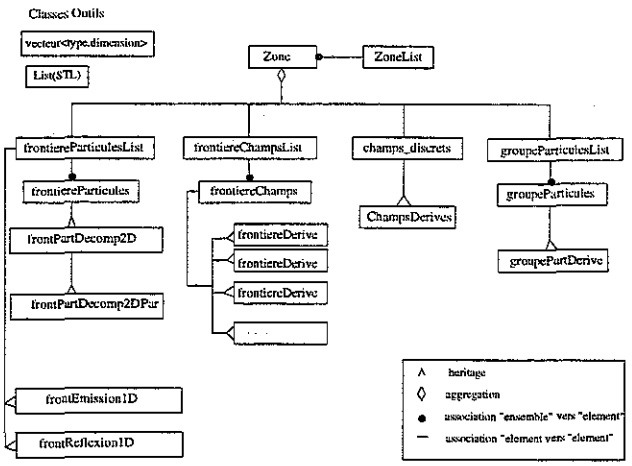


FIG. 4.1 – *Schéma général du modèle objets commun*

Les principales classes mises en jeu dans le modèle commun

- La classe "**Zone**" prend en charge la supervision de l'ensemble des opérations concernant une des zones ou la zone recouvrant l'espace traité. Elle assure le déroulement de l'algorithme principal sur un pas de temps, en utilisant pour cela les services des outils de modélisation :
  - des champs électromagnétiques réels dérivés du modèle virtuel commun,
  - des groupes de particules réels, dérivés du modèle commun,
  - des éléments "frontières sur champs" réels, dérivés du modèle virtuel commun,

- des éléments "frontières sur particules" réels, dérivés du modèle virtuel commun.

Les données et opérations traitées dans cette classe étant permanentes, les zones réelles seront instantiées directement à partir de cette classe, qui ne sera donc en principe pas dérivée. Un objet devra être instantié pour chaque zone géométrique réelle. Ces objets pourront être gérés de façon très différente selon le mode de calcul séquentiel ou parallèle, mais les fonctions internes de cette classe n'en seront pas affectées.

- La classe "**champs_discrets**" prend en charge les opérations d'échange de données minimaux entre le modèle des champs et le reste de l'application. Elle encapsule à l'aide de méthodes virtuelles, de façon à la rendre totalement transparente aux autres classes mises en cause :
  - la modélisation du calcul des champs (méthode "resolution"). La méthode de résolution fera appel aux méthodes virtuelles des classes "frontiereChamps" pour intégrer les conditions limites sur les champs à l'algorithme utilisé.
  - l'interface avec les classes prenant en charge le calcul du déplacement des groupes de particules tel que la localisation des particules dans le maillage (méthode localisation), l'interpolation des champs sur les particules (méthode champs_sur_particules) et la contribution des particules au calcul des termes sources des équations de Maxwell (méthode contribution).

  Afin de garantir une indépendance maximale, elle a été conçue comme classe conteneur devant être dérivée pour l'implémentation d'un modèle de résolution des champs électromagnétiques (type de maillage, méthode de résolution, méthode d'interpolation et de localisation). Un objet dérivé devra donc être instantié pour chaque zone, il échangera ses informations via ses propres méthodes, et devra connaitre :
  - la liste des objets dérivées "frontièresChamps" délimitant la périphérie de la zone pour la résolution des équations de Maxwell,
  - les liens existant entre la géométrie du domaine et les objets de la classe "frontiereParticules". Dans ce dernier cas, les liens sont utilisés pour gérer le transfert des particules vers l'extérieur du domaine ou vers une zone voisine, à travers un objet "frontiereParticule" et ils ne sont exploités que par la méthode virtuelle "localisation". Une modification de la liste des objets "frontiereParticules" n'implique qu'une adaptation simple de la méthode "localisation" sans modifier le reste du code.
- La classe **groupeParticules** prend en charge les opérations de modification d'état d'un groupe de particules. Elle encapsule :
  - d'une part, les opérations permanentes de maintien et d'évolution des données du groupe à l'occasion de l'évolution de la population des particules qui y sont décrites.
  - d'autre part la modélisation des déplacements des particules par la résolution des équations du mouvement de façon à les rendre totalement transparents aux autres classes mises en cause.

  Afin de garantir une indépendance maximale des modèles, elle contient une méthode virtuelle "pousse_paquet" assurant le calcul par paquet du mouvement des

particules sous l'action des champs ambiants. Cette méthode devra être surchargée par une forme polymorphe dans une classe dérivée. Un objet dérivé devra donc être instantié pour chaque type de particules présentes dans chaque zone, traitant les déplacements sur cette zone.

- La classe "**frontiereParticules** prend en charge les opérations de transfert des particules sur le domaine d'observation dans le cas d'un domaine simple ou d'une décomposition de domaine en plusieurs zones. Elle prendra en charge les échanges de données minimales entre le modèle des éléments "frontièreParticules" limitant les zones recouvrant l'espace traité. La classe encapsule la modélisation de ces éléments de façon à la rendre totalement transparente aux autres classes mises en cause, quel que soit le mode de calcul choisi, séquentiel ou parallèle. Elle modélise les frontières du domaine ayant pour seule action de laisser sortir les particules du domaine d'observation, elle devra être dérivée pour l'implémentation de comportements aux limites sur les particules différents . Elle contient deux méthodes virtuelles, l'une traitant de l'émission de particules (méthode condition_emission), l'autre de la réception des particules dans la zone courante (méthode condition_reception). Pour chaque zone, un objet devra donc être instantié pour chaque tronçon présentant un comportement particulier, selon le modèle de classe correspondant à ce comportement. Cet objet échangera ses informations via des méthodes de la classe,
  - avec les autres classes de même type modélisant la même zone,
  - avec les autres frontières des zones voisines.
- La classe **frontiereChamps** prend en charge les opérations de traitement des conditions limites sur les champs dans le cas, d'un domaine simple ou d'une décomposition de domaine en plusieurs "zones". Elle prendra en charge les échanges de données minimales entre le modèle des éléments "frontièreChamps" limitant les zones recouvrant l'espace traité. Dans cette classe seront traités uniquement *les échanges de données sur les champs*, les échanges sur les particules étant traitées par les classes dérivées de la classe "**frontiereParticules**". La classe encapsule la modélisation de ces éléments de façon à la rendre totalement transparente aux autres classes mises en cause, quel que soit le mode de calcul choisi, séquentiel ou parallèle. Afin de garantir une indépendance maximale, elle a été conçue comme classe conteneur devant être dérivée pour l'implémentation des différents comportements aux limites. Elle contient une méthode virtuelle "conditions_limites" pour l'intégration des conditions limites à l'algorithme de résolution des équations de Maxwell. Pour chaque zone, un objet devra donc être instantié pour chaque tronçon présentant un comportement particulier, selon le modèle de classe correspondant à ce comportement. Cet objet échangera ses informations via les méthodes de la classe,
  - avec les autres classes modélisant la même zone.
  - avec d'autres éléments "frontièreChamps" des zones voisines.

Pour des raisons de performances, les particules en transit vers une zone voisine seront transmises par paquet une seule fois par itération en temps. C'est la classe "zone" qui gére ce transit par l'intermédiaire de deux buffers, l'un stockant les particules sortantes triées par élément "frontiereParticle", l'autre les particules entrantes triées par type de particule

et des méthodes (condition_emission, condition_sortant) qui servent d'interface avec les classes "frontiereParticules" chargées du transfert.

Finalement, la classe "**zone**", chef d'orchestre des différentes opérations a pour composantes, un objet de la classe "champs_discrets", un tableau ou une liste d'objets de la classe "groupeParticules" , un tableau ou une liste d'objets de la classe "frontiereParticule", un tableau ou une liste d'objets de la classe "frontiereChamps" ainsi que deux buffers, "buffer_sortant" trié par type de "frontiereParticule" et "buffer_entrant" trié par type de particules permettant le stockage provisoire des particules en transit.

Traitement des particules entrantes et sortantes

Après le calcul du déplacement des particules, la gestion des particules sortantes ou en transit se fait en trois étapes à chaque itération en temps :

1. La mise à jour par chaque objet "groupeParticule" du buffer_sortant de l'objet "zone" associé. Elle est prise en charge par la méthode "localisation" de cet objet.
2. L'émission des particules sortantes en provenance de deux sources possibles, l'émission de particules vers une zone voisine ou vers l'extérieur du domaine et la réflexion de particules sur l'élément courant. Elle se fait globalement sur la zone par la méthode "condition_emission" de la classe "zone" qui fait appel à la méthode virtuelle "condition_emission" de la classe "frontiereParticules". Cette dernière effectue l'envoi par paquets des particules sortantes à l'élément "frontiereParticules" destinataire. Elles est exécutée par tous les objets "frontiereParticules" de la zone même si aucune particule ne sort par cet élément. C'est cet appel qui forcera la synchronisation des processeurs dans le cas d'une implémentation parallèle.
   Trois cas sont envisagés :
   - Dans le cas du calcul sur une seule zone, elle consiste à éliminer les particules stockées dans buffer_sortant. C'est aussi à ce niveau que sera traitée la réflexion, en stockant dans le buffer_entrant de la zone courante les particules réfléchies.
   - Dans le cas d'une décomposition de domaine sur un seul processeur, seule le cas de la réflexion est à traiter. Le transfert vers buffer_entrant des particules sortantes se fera à la reception.
   - Dans le cas d'une décomposition de domaine sur plusieurs processeurs, elle consiste à envoyer le paquet de particules à l'objet "frontiereParticules" associé de la zone voisine. L'objet "frontiereParticules" concerné doit connaitre cet objet mais il devra connaitre également l'identificateur du processeur qui le traite.
3. La réception des particules entrantes en provenance de deux sources possibles, la réception de particules venant d'une zone voisine et l'émission physique par l'élément courant. Elle se fait globalement sur la zone par la méthode "condition_reception" de la classe "zone" qui fait appel à la méthode virtuelle "condition_reception" de la classe "frontiereParticules". Cette dernière effectue la réception par paquets des particules entrantes de l'élément "frontiereParticules" émetteur. Elles est exécutée

par tous les objets "frontiereParticules" de la zone même si aucune particule n'entre par cet élément. C'est cet appel qui forcera la synchronisation des processeurs dans le cas d'une implémentation parallèle.
Trois cas sont envisagés :

- Dans le cas du calcul sur une seule zone, elle ne concerne que l'émission physique de particules par l'élément "frontiereParticules" et consiste à stockér les particules émises dans buffer_sortant de l'objet "zone" associé.
- Dans le cas d'une décomposition de domaine sur un seul processeur, elle consiste à transferrer les particules stockées dans buffer_sortant de la zone émettrice dans le buffer_entrant de la zone receptrice. Elle peut concerner également l'émission physique de particules.
- Dans le cas d'une décomposition de domaine sur plusieurs processeurs. Elle consiste à recevoir le paquet de particules envoyé par la zone émettrice et à les stocker dans le buffer_entrant de la zone receptrice. Elle peut concerner également l'émission physique de particules.

L'implémentation des trois configurations précitées se fera par dérivation successives de la classe "frontiereParticules".

Traitement d'une itération en temps

Le *traitement d'un pas de temps* est pris en charge par la méthode "**deroulement_temps**" de chaque classe zone. Chaque itération effectue les opérations suivantes :

1. La réception des particules entrantes.
2. Le calcul des termes sources des équations de Maxwell dépendent de la densité de courant et de la densité de charges recalculées après le déplacement des particules sur la position de ces particules. Ces quantités seront calculées par interpolation sur les points du maillage par la méthode "**contribution**" de la classe **groupeParticules** qui fera appel à la *méthode virtuelle* **contribution** de la classe **champs_discrets**. Le type d'interpolation utilisée se fera à l'implémentation de cette dernière dans la classe dérivée de la classe **champs_discrets**.
3. Le calcul des champs électromagnétiques se fait par la *méthode virtuelle* **résolution** de la classe **champs_discrets**. Elle calcule les champs sur les noeuds du maillage en fonction des termes sources calculés précédemment et les conditions aux limites implémentées par les classes dérivées de la classe **frontiereChamps**.
4. Le calcul du déplacement des particules à l'aide de l'équation du mouvement par la méthode virtuelle **pousse_particules** sur tous les objets "**groupeParticules** de la zone. Les valeurs des champs agissant sur les particules sont calculés par des appels à la *méthode virtuelle* "**champs_sur_particules**" de "**champs_discrets**".
5. Avant de commencer l'itération suivante, il sera nécessaire :
   - de localiser les particules dans le maillage par la méthode "**localisation**" sur tous les "**groupeParticules**" de la zone qui fera appel à la *méthode virtuelle* "**localisation**" de la classe "**champs_discrets**". Toutes les particules sortant de la zone sont triées par élément "**frontiereParticules**" destinataire et stockées dans un buffer_sortant, composante de la classe "zone" courante.

- d'émettre les particules sortantes, par la méthode "**condition_emission**" de la classe zone qui lancera la méthode "**condition_emission**" sur tous les éléments "**frontiereParticules**" référencés dans la zone. Le cas de la réflexion des particules sortantes est traité à ce niveau.

Quelques remarques

1. La classe "zone" sert uniquement à gérer l'interface entre les problèmes de résolutions des champs électromagnétiques et du déplacement des particules sous l'effet de ces champs.
2. L'intéraction entre les classes "groupeParticules" et "champs_discrets" se fait par l'intermédiaire de méthodes de ces deux classes.
   - Le calcul des intéractions des particules et des champs nécessite de connaitre la position des particules par rapport aux mailles. La méthode **localisation** de la classe "groupeParticules" fournit à la méthode virtuelle "**localisation**" de la classe "champs_discrets" un ensemble de particules connues par leur position et leur vitesse et récupère, pour celles qui quittent la zone, l'information nécessaire à leur transfert, et pour les autres, leur localisation (numéro de maille) dans la zone. A chaque pas de temps, pour chaque objet "groupeParticules", elle sera appelée une fois après déplacement des particules internes, et une fois pour localiser les nouvelles particules pénétrant dans la zone. Cette méthode est totalement indépendante du cas test traité, c'est l'implémentation de la méthode virtuelle "localisation" d'une classe dérivé de la classe "champs_discrets" qui va permettre une adaptation de notre modèle au type de maillage utilisé et aux liens directs reliants la géométrie des champs à celle des éléments frontières.
   - La méthode "**contribution**" de la classe "groupeParticules" permet d'informer le modèle des champs de la présence d'un ensemble de particules connues par leur charge, leur position et leur vitesse. La méthode virtuelle "**contribution**" de la classe "champs_discrets" est ainsi en mesure de calculer les termes sources des équations de Maxwell qu'elle est chargé de résoudre. Cette méthode sera appelée une fois pour chaque objet "groupeParticules", à chaque pas de temps. La méthode "**contribution** de la classe "groupeParticules" est totalement indépendante du cas test traité, c'est l'implémentation de la méthode virtuelle "contribution" d'une classe dérivé de la classe "champs_discrets" qui va permettre une adaptation de notre modèle aux spécificités du cas test traité.
   - La méthode **champs_sur_particules** de la classe "champs_discrets" est appelée par une méthode de la classe "groupeParticules", pour calculer les champs sur la position des particules, afin d'en évaluer le déplacement au cours de chaque pas de temps. Comme précédemment, ce calcul, propre au modèle de représentation choisi, sera pris en charge par polymorphisme.

   Pour ces trois méthodes, l'échange est effectué par paquets pour minimiser l'overhead du à un nombre trop grand d'appels.

4.4 Implémentation d'un code PIC

4.4.1 Introduction

La description du modèle objet commun d'un code PIC s'est faite à partir du modèle mathématique indépendament du modèle discrétisé. Le but de ce paragraphe est de décrire l'implémentation d'un modèle discrétisé en 2D utilisant les méthodes standards de discrétisation et de résolution étudiées dans ([11]). Ce n'est pas une étude complète de la stabilité et des propriétés de convergence de l'algorithme choisi mais l'adaptation d'un cas test de la physique des plasmas utilisant une méthode particulaire à partir du modèle objet commun présenté précédemment. Le code ainsi construit sera utilisé dans le prochain chapitre pour la mise en oeuvre de l'implémentation nécessaire à l'étude de plusieurs méthodes de correction du champs électrique dans un code PIC.

Nous avons montré que l'implémentation d'un modèle discrétisé consiste à la création de classes dérivées des classes de base du modèle commun :

- *une classe maillage* définissant la discrétisation du domaine de calcul des champs électromagnétiques,
- une classe dérivée de la classe "champs_discrets", ayant pour composante une objet "maillage" et une implémentation des méthodes virtuelles de résolution des champs, d'interpolation des champs sur les particules et de contribution des particules aux termes sources des équations de Maxwell,
- *des classes dérivées des classes "frontiereChamps"* permettant la prise en charge dans la résolution, des conditions limites sur les champs,
- *des classes dérivées des classes "frontiereParticules"* pour l'implémentation du transfert des particules entre les zones ou vers l'extérieur du domaine et éventuellement l'emission et la réflexion de particules à partir d'un de ces élément,
- *une classe dérivée de la classe "groupeParticle"* fournissant l'implémentation de la résolution de l'équation du mouvement pour la mise à jour de la vitesse et de la position des particules.

La classe "Zone" ne sera pas dérivée, elle jouera le rôle de chef d'orchestre entre les différentes classes du modèle.

Dans un premier temps, pour simplifier la description des méthodes utilisées, on se place dans le cas d'une seule zone, on décrira ultérieurement les modifications apportées pour une résolution sur plusieurs zones par une décomposition de domaine.

4.4.2 Algorithme

Intégration en temps des champs et localisation de leurs composantes

En dimension 2, les champs peuvent être décomposés en composantes électriques transverses (TE) et en composantes magnétiques transverses (TM), toutes les variations spatiales se font dans le plan (x,y). Les champs TE ont pour composantes (Ez, Bx, By) et les champs TM, (Ex, Ey, Bz). En reportant dans les équations (4.1), on constate que ces deux

groupes de composantes sont non couplés. Dans certains cas les composantes électriques transverses (TE) restent nulles, on ne calculera alors que les composantes magnétiques transverse (TM). C'est le cas lorsque les composantes Ez, Bx, By et vz sont nulles à l'instant t=0, on se placera dans cette situation.

On effectue une discrétisation structurée régulière du domaine d'observation (nx points le long de l'axe x, ny le long de l'axe y) et on choisit la localisation des composantes des champs (Ex, Ey, Bz) ainsi que celle des termes sources des équations de Maxwell (ρ, Jx, Jy) sur le maillage de telle façon que les schémas aux différences finies en espace utilisés soient d'ordre 2. Pour satisfaire à des critères de performance (minimiser les tests sur la position des particules dans les calculs d'interpolation), on ajoutera autour du domaine de calcul des points fictifs (Fig:4.2).

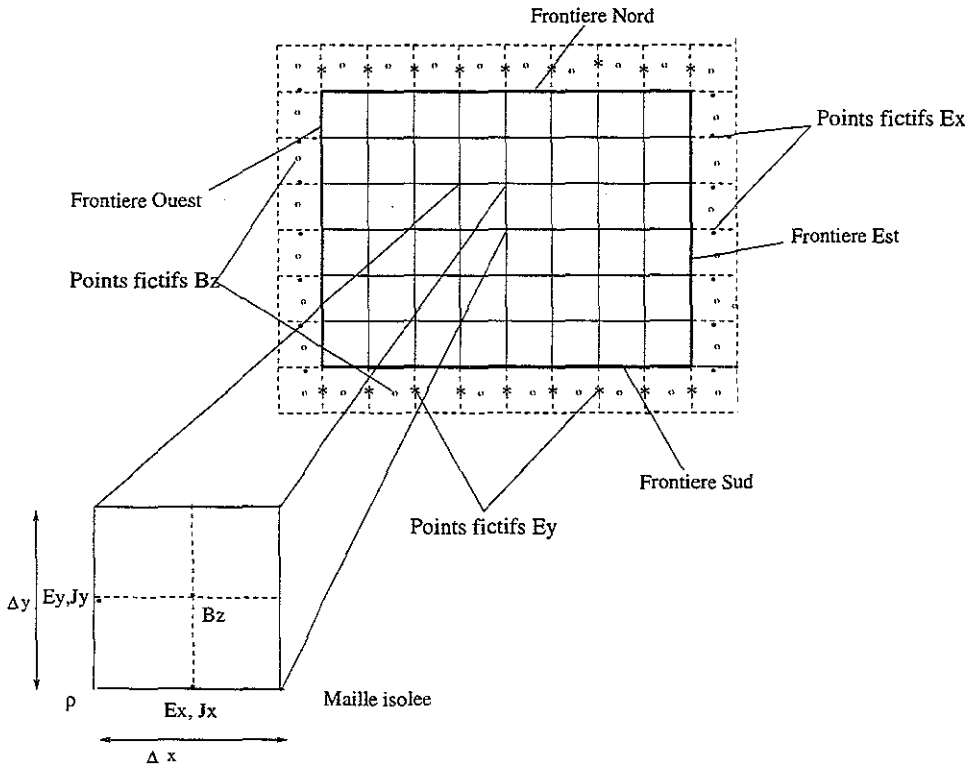


FIG. 4.2 – *Schéma de discrétisation du domaine d'observation*

Résolution des équations de Maxwell avec conditions limites de Silver Muller

On utilise le shéma d'ordre deux en temps et en espace proposé par ([11]).

1. Calcul du champ magnétique au temps $n+\frac{1}{2}$ sur les points $(j+\frac{1}{2},k+\frac{1}{2})$ intérieurs au maillage :

   On utilise l'équation de Faraday du système (4.1) pour calculer $B^{n+\frac{1}{2}}_{z,j+\frac{1}{2},k+\frac{1}{2}}$. Pour des raisons d'ordre d'approximation, le calcul de la position des particules au temps $n+\frac{1}{2}$ utilise B^n, afin éviter le stockage de $B^{n-\frac{1}{2}}$ et B^n, le calcul se fera sur deux

demis pas de temps,

- un demi pas de temps avant le déplacement des particules pour le calcul de B_z^n

$$B_z^n = B_z^{n-\frac{1}{2}} - \frac{\Delta t}{2}\nabla \times E^n,$$

- puis un deuxième demi pas de temps à l'itération suivante pour le calcul de $B_z^{n+\frac{1}{2}}$

$$B_z^{n+\frac{1}{2}} = B_z^n - \frac{\Delta t}{2}\nabla \times E^n.$$

On conserve ainsi l'ordre du schéma de discrétisation

$$B^{n+\frac{1}{2}}_{z,j+\frac{1}{2},k+\frac{1}{2}} = B^n_{z,j+\frac{1}{2},k+\frac{1}{2}} - \frac{\Delta t}{2}\frac{E^n_{y,j+1,k+\frac{1}{2}} - E^n_{y,j,k+\frac{1}{2}}}{\Delta x} + \frac{\Delta t}{2}\frac{E^n_{x,j+\frac{1}{2},k+1} - E^n_{x,j+\frac{1}{2},k}}{\Delta y}.$$

2. Prise en compte des conditions limites sur les champs :
On impose les conditions de Silver-Muller (pas de champs entrants) sur les frontières Nord, Sud, Est et Ouest du domaine (Fig :4.2).

$$\begin{cases} E_y &= cB_z \quad \text{sur} \quad \Gamma_{Ouest}, \\ E_y &= -cB_z \quad \text{sur} \quad \Gamma_{Est}, \\ E_x &= cB_z \quad \text{sur} \quad \Gamma_{Nord}, \\ E_x &= -cB_z \quad \text{sur} \quad \Gamma_{Sud} \end{cases} \tag{4.3}$$

où le coefficient "c" est la vitesse de la lumière.

On effectue le calcul du champ magnétique sur les points fictifs du maillage simultanément avec la prise en compte des conditions limites sur les champs électriques. Pour cela, on résout sur les points frontières, l'équation de la condition limite en moyennant en temps pour le champ électrique et en espace pour le champ magnétique puis l'équation d'Ampère. Ce qui nous amène à résoudre le système suivant pour la frontière Nord :

$$\begin{cases} E^{n+1}_{x,j+\frac{1}{2},n_y-1} + E^n_{x,j+\frac{1}{2},n_y-1} &= -c(B_{z,j+\frac{1}{2},n_y-\frac{1}{2}} + B_{z,j+\frac{1}{2},n_y-\frac{3}{2}}), \\ (\partial t E_x - c^2 \partial x B_z)^{n+\frac{1}{2}}_{j+\frac{1}{2},n_y-1} &= -\frac{1}{\epsilon_0} J^{n+\frac{1}{2}}_{x,j+\frac{1}{2},n_y-1}. \end{cases} \tag{4.4}$$

Ce qui nous donne :

$$\begin{cases} E^{n+1}_{x,j+\frac{1}{2},n_y-1} &= (1 - \frac{2c\Delta t}{\Delta y + c\Delta t})E^n_{x,j+\frac{1}{2},n_y-1} - \frac{2c^2\Delta t}{\Delta y + c\Delta t}B^{n+\frac{1}{2}}_{z,j+\frac{1}{2},n_y-\frac{3}{2}} \\ & \quad -0.5\frac{\Delta t \Delta y}{\epsilon_0(\Delta y - c\Delta t)}J^{n+\frac{1}{2}}_{x,j+\frac{1}{2},n_y-1}, \\ B^{n+\frac{1}{2}}_{z,j+\frac{1}{2},n_y-\frac{1}{2}} &= -\frac{2\Delta y}{c(\Delta y + c\Delta t)}E^n_{x,j+\frac{1}{2},n_y-1} + (-1 + \frac{2c\Delta t}{\Delta y + c\Delta t})B^{n+\frac{1}{2}}_{z,j+\frac{1}{2},n_y-\frac{3}{2}} \\ & \quad -0.5\frac{\Delta t \Delta y}{\epsilon_0 c(\Delta y - c\Delta t)}J^{n+\frac{1}{2}}_{x,j+\frac{1}{2},n_y-1}. \end{cases}$$

Nous appliquons le même principe de calcul pour les frontières Sud, Est et Ouest.

3. Calcul des composantes du champ électrique au temps n+1 sur les points internes du maillage :
On utilise l'équation d'Ampère du système (4.1) pour calculer

$$E_x^{n+1} \text{ aux points } (j+\tfrac{1}{2},k),$$
$$E_y^{n+1} \text{ aux points } (j,k+\tfrac{1}{2}).$$

On a les équations

$$\begin{cases} (\partial t E_x)_{j+\frac{1}{2},k}^{n+\frac{1}{2}} &= (c^2\partial_y B_z - \dfrac{1}{\epsilon_0} J_x)_{j+\frac{1}{2},k}^{n+\frac{1}{2}}, \\ (\partial t E_y)_{j,k+\frac{1}{2}}^{n+\frac{1}{2}} &= (-c^2\partial_x B_z - \dfrac{1}{\epsilon_0} J_y)_{j,k+\frac{1}{2}}^{n+\frac{1}{2}}. \end{cases} \tag{4.5}$$

Ce qui nous donne :

$$\begin{cases} E_{x,j+\frac{1}{2},k}^{n+1} &= E_{x,j+\frac{1}{2},k}^{n} + \dfrac{c^2\Delta t}{\Delta y}(B_{z,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} - B_{z,j+\frac{1}{2},k-\frac{1}{2}}^{n+\frac{1}{2}}) - \dfrac{\Delta t}{\epsilon_0} J_{x,j+\frac{1}{2},k}^{n+1}, \\ E_{y,j,k+\frac{1}{2}}^{n+1} &= E_{y,j,k+\frac{1}{2}}^{n} - \dfrac{c^2\Delta t}{\Delta y}(B_{z,j+\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}} - B_{z,j-\frac{1}{2},k+\frac{1}{2}}^{n+\frac{1}{2}}) - \dfrac{\Delta t}{\epsilon_0} J_{y,j,k+\frac{1}{2}}^{n+1}. \end{cases}$$

Il sera nécessaire de faire la mise à jour par interpolation du champ électrique sur les points fictifs.

Interpolation

- Calcul de la densité de charge au noeud j du maillage :

$$\rho_j = \sum_{i(particules)} p_i q_i S(X_j - x_i) \tag{4.6}$$

où S est la fonction d'interpolation, p_i le poids associé à la particule, q_i la charge de la particule, X_j la position du noeud j et x_i la position de la particule i. Selon le choix de la fonction d'interpolation, l'approximation de ρ sera d'ordre 0, 1 ou 2 (rarement d'ordre supérieure).

- Calcul de la densité de courant au temps $n+\frac{1}{2}$: deux méthodes peuvent être envisagées,
  - sur les particules qui passent la frontière au temps $n+\frac{1}{2}$

$$J_j^{n+\frac{1}{2}} = \sum_{i(particules)} V_i^{n+\frac{1}{2}} \frac{S(X_j - x_i^n) + S(X_j - x_i^{n+1})}{2}, \tag{4.7}$$

  - ou en deux étapes,

$$\begin{cases} x_i^{n+\frac{1}{2}} &= x^n + v_i^{n+\frac{1}{2}} \times \dfrac{\Delta t}{2}, \\ J_j^{n+\frac{1}{2}} &= \displaystyle\sum_{i(particules)} V_i^{n+\frac{1}{2}} \times S(X_j - x_i^{n+\frac{1}{2}}). \end{cases} \tag{4.8}$$

Le coût de ces deux méthodes est identique, nous avons implémenté la seconde.

- Calcul des forces provenant du champ électrique (F_i) et du champ magnétique (G_i) sur une particule :

$$F_i = \frac{q_i}{m_i} \sum_{j(noeuds)} E_j \times S(X_j - x_i), \tag{4.9}$$

$$G_i = \frac{q_i}{m_i} \sum_{j(noeuds)} B_j \times S(X_j - x_i). \tag{4.10}$$

On a utilisé la même fonction S dans (4.6, 4.8, 4.9, 4.10). On pourrait utiliser des fonctions d'interpolation différentes pour (4.6) et (4.9, 4.10), mais l'utilisation de la même fonction d'interpolation élimine les "self-force" et assure la conservation des moments.Les différentes méthodes d'interpolation peuvent être implémentées dans notre application par dérivation de la classe chargé de la résolution des champs électromagnétiques, à l'aide soit de la *méthode virtuelle* "contribution" en ce qui concerne le calcul des termes sources des équations de Maxwell, soit de la *méthode virtuelle* "champ_sur_particules" pour le calcul des forces issues des champs électromagnétiques exercées sur les particules.

Mouvement des particules

Le calcul du déplacement des particules sous l'action des champs électromagnétiques se fait en deux étapes.

1. Interpolation des champs sur les particules :

Avant l'interpolation des champs sur les particules, il est nécessaire de calculer B^{n+1} sur le deuxième demi pas de temps.

$$B^{n+1} = B^{n+\frac{1}{2}} - \frac{\Delta t}{2} \nabla \times E^n$$

Ce qui nous donne sur les points du maillage de B

$$B^{n+1}_{z,j+\frac{1}{2},k+\frac{1}{2}} = B^{n+\frac{1}{2}}_{z,j+\frac{1}{2},k+\frac{1}{2}} - \frac{\Delta t}{2} \frac{E^n_{y,j+1,k+\frac{1}{2}} - E^n_{y,j,k+\frac{1}{2}}}{\Delta y} + \frac{\Delta t}{2} \frac{E^n_{x,j+\frac{1}{2},k+1} - E^n_{x,j+\frac{1}{2},k}}{\Delta x}.$$

Pour le calcul de l'interpolation des champs sur les particules, il sera plus simple d'interpoler d'abord E et B sur les points du maillage de ρ de manière à n'avoir qu'une seule grille pour l'interpolation sur les particules. L'interpolation sur les particules se fera donc en deux phases. L'implémentation se fait par une méthode virtuelle "champs_sur_particules" d'une classe dérivée de la classe "champs_discrets". Une modification de la méthode d'interpolation consiste à modifier l'implémentation de cette méthode par dérivation sans aucune autre modification ou répercussion dans le reste du code.

2. Calcul de la vitesse des particules par l'intégration en temps de l'équation du mouvement :

$$\begin{cases} m\dfrac{dv}{dt} &= q(E + v \wedge B), \\ \dfrac{dx}{dt} &= v. \end{cases} \tag{4.11}$$

On suppose que E^n et B^n sont interpolés sur les particules à partir des noeuds du maillage de ρ à l'instant $t^n = n\Delta t$. Pour une généralisation relativiste, on utilise $u = \gamma v$ avec $\gamma^2 = 1 + \dfrac{u^2}{c^2}$, on obtient :

$$\frac{u^{n+\frac{1}{2}} - u^{n-\frac{1}{2}}}{\Delta t} = \frac{q}{m}(E^n + \frac{u^{n+\frac{1}{2}} + u^{n-\frac{1}{2}}}{2\gamma^n} \wedge B^n)$$

avec $(\gamma^n)^2 = 1 + \dfrac{(u^n)^2}{c^2}$.

On utilise la méthode de séparation des effets électriques et magnétiques qui permet d'éviter une accélération non physique des particules par le champ magnétique. Il s'agit de séparer les forces provenant du champ magnétique $(\frac{q}{m}v \wedge B)$ et celle provenant du champ électrique $(\frac{q}{m}E)$ ([11]). Le calcul se fait en trois phases :

(a) On ajoute la moitié de l'effet de E :

$$\begin{cases} u^{n-\frac{1}{2}} &= u_1^{n+\frac{1}{2}} - \dfrac{qE^n}{m}\dfrac{\Delta t}{2}, \\ u^{n+\frac{1}{2}} &= u_2^{n+\frac{1}{2}} + \dfrac{qE^n}{m}\dfrac{\Delta t}{2}. \end{cases} \tag{4.12}$$

On en déduit :

$$u_1^{n+\frac{1}{2}} = u^{n-\frac{1}{2}} + \frac{qE^n}{m}\frac{\Delta t}{2}.$$

(b) On effectue ensuite la rotation due au champ magnétique, on obtient $u_2^{n+\frac{1}{2}}$ avec

$$\begin{cases} \dfrac{u_2^{n+\frac{1}{2}} - u_1^{n+\frac{1}{2}}}{\Delta t} &= \dfrac{q}{m}(\dfrac{u_2^{n+\frac{1}{2}} + u_2^{n+\frac{1}{2}}}{2\gamma^n}) \wedge B^n. \end{cases} \tag{4.13}$$

$u_2^{n+\frac{1}{2}}$ est l'image de $u_1^{n+\frac{1}{2}}$ par la rotation d'axe parallèle à B^n d'angle θ^n,

$$\theta^n = -2\arctan(\frac{qB_z^n\Delta t}{2\gamma^n m})$$

avec $(\gamma^n)^2 = 1 + (\dfrac{u_1^{n+\frac{1}{2}}}{c})^2$.

On utilise l'algorithme de Buneman ([11]) qui optimise le nombre d'opérations à effectuer pour le calcul de θ^n.L'algorithme est le suivant :

$$\left\{\begin{array}{lcll} t & = & -\tan\dfrac{\theta^n}{2} & = \dfrac{B_z^n \Delta t}{2\gamma^n} \times \dfrac{q}{m} \\ s & = & -\sin\theta^n & = \dfrac{2t}{1+t^2} \\ c & = & \cos\theta^n & = \dfrac{1-t^2}{1+t^2} \end{array}\right. \tag{4.14}$$

$$\left\{\begin{array}{lcl} u_x^{n+\frac{1}{2}'} & = & u_{1,x}^{n+\frac{1}{2}} + u_{1,y}^{n+\frac{1}{2}} \times t \\ u_{2,y}^{n+\frac{1}{2}} & = & u_{1,y}^{n+\frac{1}{2}} - u_x^{n+\frac{1}{2}'} \times s \\ u_{2,x}^{n+\frac{1}{2}} & = & u_x^{n+\frac{1}{2}'} u_{2,y}^{n+\frac{1}{2}} \times t. \end{array}\right. \tag{4.15}$$

(c) On ajoute enfin la moitié restante due à l'effet du champ électrique

$$\left\{ u^{n+\frac{1}{2}} = u_2^{n+\frac{1}{2}} + \frac{q}{m} E^n \frac{\Delta t}{2\gamma^n}. \right. \tag{4.16}$$

3. Calcul de la position de la particule par :

$$\left\{\begin{array}{lcl} x^{n+1} & = & x^n + v^{n+\frac{1}{2}} \Delta t, \\ v^{n+\frac{1}{2}} & = & \dfrac{u^{n+\frac{1}{2}}}{\gamma^{n+\frac{1}{2}}}, \\ \gamma^{n+\frac{1}{2}} & = & 1 + \dfrac{u^{n+\frac{1}{2}}}{v^{n+\frac{1}{2}}}. \end{array}\right. \tag{4.17}$$

Traitement des particules entrantes et sortantes

Si on veut intégrer une émission de particules par l'une des frontières du domaine, il sera nécessaire de construire une classe dérivée de la classe "frontiereParticules", elle sera chargée de l'émission et de la réception des particules. Le traitement des particules sortantes ou en transit est intégralement récupéré du modèle commun quelque soit le modèle de calcul, séquentiel ou parallèle.

4.4.3 Implémentation

Sur une seule zone, l'implémentation est immédiate, nous avons vu qu'il s'agissait d'implémenter (Fig :4.3) :

1. Une classe dérivée "champs2D" de la classe "champs_discrets" dans laquelle on implémente :
   - La méthode "contribution" appelée par la méthode "contribution" de la classe "groupeParticules". Elle effectue par interpolation linéaire, à partir de la position des particules, le calcul des densités de charges et de courant sur les noeuds du maillage de résolution des champs. La méthode étant virtuelle, un changement de fonction d'interpolation se fera par dérivation et implémentation de la nouvelle fonction, sans aucune autre modification dans le code.

- La méthode "resolution" qui calcule les champs électromagnétiques par l'algorithme décrit précédemment. Elle fait appel à la méthode "conditions_limites" de tous les objets dérivés de "frontiereChamps" de la zone.
- La méthode "champs_sur_particules" qui effectue l'interpolation des champs sur les particules après la résolution des équations de Maxwell.
- La méthode "localisation" localisant les particules dans le maillage après le calcul de leur déplacement par résolution des équations de mouvement.

2. Une classe dérivée de la classe "frontiereChamps" par type de comportement aux limites des champs. Nous avons quatre types de conditions limites sur les champs donc quatre classes dérivés : "frontiereChampsSud", "frontiereChampsNprd", "frontiereChampsOuest", "frontiereChampsEst".
3. Une classe dérivée de la classe "groupeParticules" dans laquelle, on implémente la méthode "pousse_particules" chargée de résoudre l'équation du mouvement suivant l'algorithme décrit précédemment. Seule l'implémentation de la méthode pousse_particules a été nécessaire.
4. Une classe "frontiereParticulesEmission" modélisant l'émission de particules. Pour gérer le transfert des particules à travers les frontieres du domaine ou des sous-domaines, on utilisera la classe "frontiereParticules" du modèle commun.

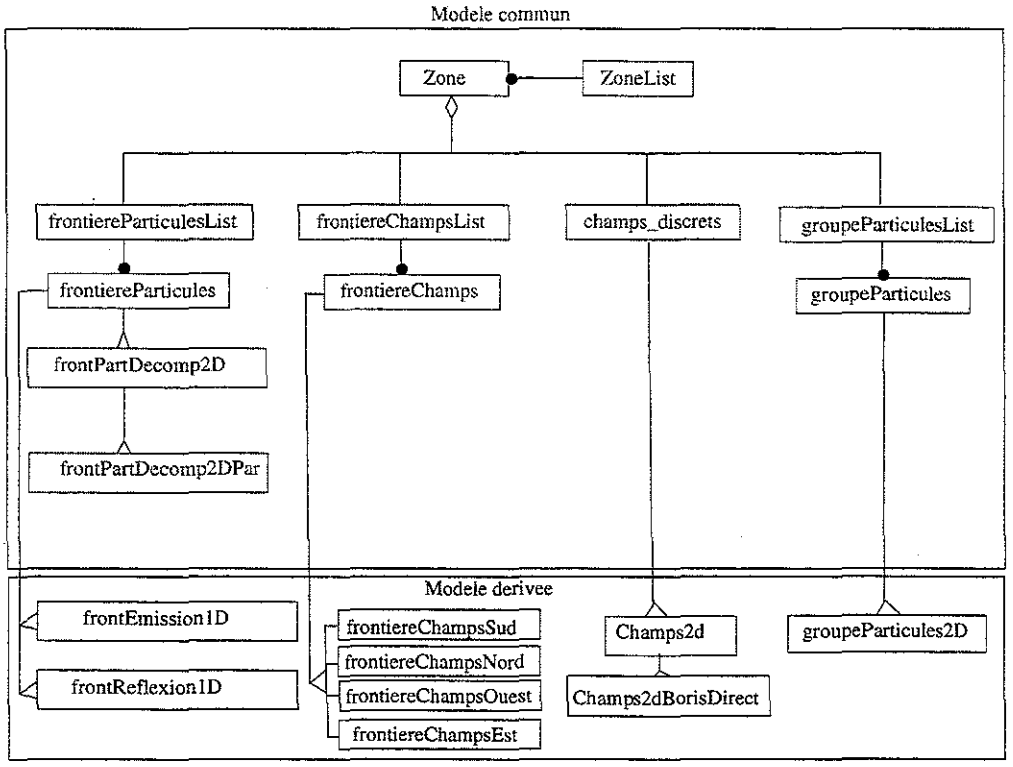


FIG. 4.3 – *Modèle dérivé adapté à notre cas test*

Remarques :

La conception du modèle objet, totalement adaptée à une méthode de résolution par décomposition de domaine permet une implémentation parallèle de type SPMD de façon assez naturelle.

En effet :

- Chaque "zone" de la décomposition est traitée à partir du modèle décrit précédemment, l'échange entre les zones s'effectuant par l'intermédiaire des éléments "frontiereChamps" et "frontiereParticules".
- La parallélisation des différentes parties du code, résolution des champs électromagnétiques, calcul du déplacement des particules et traitement des particules en transit, est totalement indépendante.
- En ce qui concerne le transfert des particules à travers les différentes zones, la conception du traitement des particules entrantes et des particules sortantes dans le modèle commun permet de localiser la parallélisation dans l'implémentation des méthodes virtuelles "condition_reception" et "condition_emission" des différents éléments dérivés de la classe "frontiereParticules". Dans le cas du traitement des communications par envoi de messages, la méthode "condition_emission" envoie au processeur concerné, l'ensemble des particules stockées dans la composante "buffer_sortant" de l'objet "zone" et la méthode "condition_reception" reçoit les particules entrantes des processeurs voisins et les stocke par type de particules dans la composante "buffer_entrant" de l'objet "zone". Il est important de remarquer qu'il n'y a aucune modification à effectuer sur le code de traitement séquentiel, en ce qui concerne la mise à jour de ces deux buffers, partie la plus délicate du traitement des particules en transit.
- Pour le traitement de la résolution des champs électromagnétiques, dans le cas d'une discrétisation en temps explicite des équations de Maxwell, une simple décomposition de domaine avec recouvrement des zones voisines sur une maille (Fig :4.4) permet de traiter chaque zone avec le modèle séquentiel. L'échange des informations sur les zones de recouvrement entre les zones voisines se fait à chaque pas de temps par les éléments frontières du modèle qui seront eux spécifiques à l'implémentation parallèle de l'application. Les éléments "frontièreParticules" et "frontiereChamps" ne correspondent pas, dans cette configuration, aux mêmes frontières géométriques des zones (Fig :4.4) . Dans le cas d'une discrétisation en temps plus implicite, l'implémentation de la méthode de résolution des champs, d'une classe dérivée de la classe "champ_discret", devra prendre en charge l'algorithme de parallélisation.

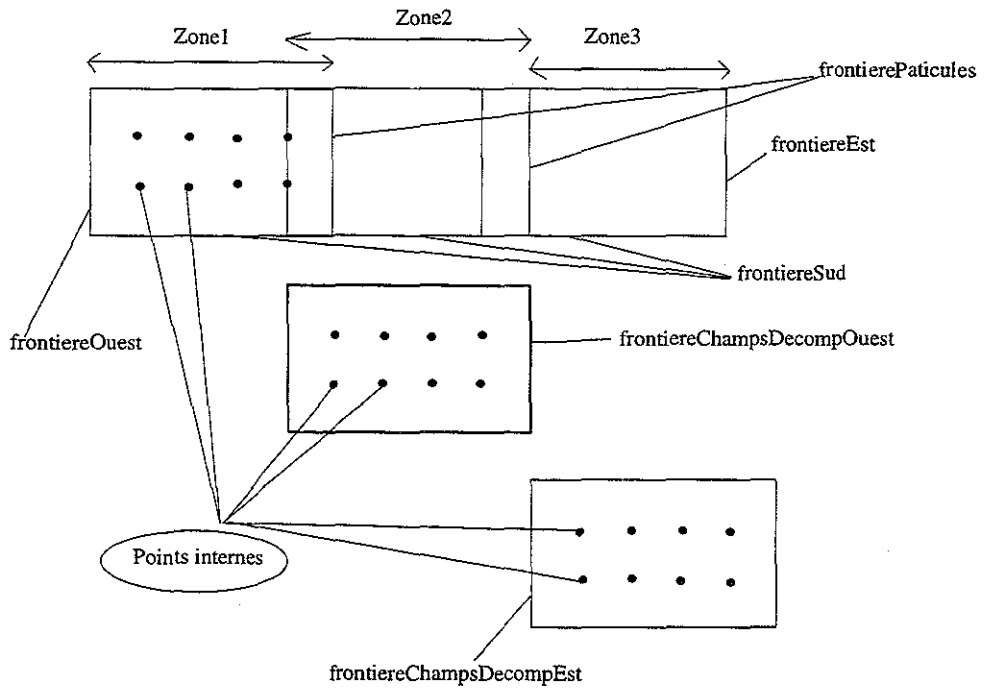


FIG. 4.4 – *Discretisation et décomposition de domaine : modèle parallèle SPMD*

Chapitre 5

Correction du champ électrique dans un code PIC

5.1 Introduction

Dans une simulation utilisant une méthode PIC comme dans beaucoup d'autres simulations numériques, des approximations doivent être faites pour permettre la simulation numérique de problèmes physiques dans un temps raisonnable. Ces approximations peuvent induire des résultats qui s'éloignent de la physique. Les approximations qui sont couramment inclues par le schéma numérique dans un modèle PIC ([11]) sont : la dicrétisation en espace du domaine d'observation par un maillage pour résoudre les équations de Maxwell; l'interpolation à partir des informations provenant des particules, des densités de charge et de courant sur les noeuds du maillage; l'interpolation des champs connus sur les noeuds du maillage sur les particules pour déterminer leur déplacement; ainsi que le nombre de particules souvent très inférieur au nombre de particules du système physique réel. Toutes ces approximations génèrent des erreurs, le but est de les minimiser de telle façon que la simulation ne s'écarte pas trop du problème physique.

Dans ce chapitre, nous étudions différentes méthodes permettant de corriger les erreurs provenant principalement du calcul des densités de charge et de courant sur les noeuds du maillage de résolution des équations de Maxwell en vue de calculer les termes sources de l'équation d'Ampère (5.1). Ce calcul est effectué par interpolation des informations provenant des particules, l'équation de conservation de la charge n'est alors pas satisfaite de façon exacte, ce qui entraine un écart à la loi de Poisson ($\nabla.E = \frac{1}{\epsilon_0}\rho$) qui peut devenir arbitrairement grand sur des temps longs (voir figure 5.1). Nous considérons deux approches : la première consiste à faire un calcul des densités par un algorithme plus compliqué tenant compte du déplacement des particules de telle façon que l'équation de conservation de la charge soit vérifiée. Cette approche nous fournira une solution de référence; la seconde approche consiste à effectuer une correction des champs électriques après la résolution des équations de Maxwell. On comparera les solutions obtenues à partir des méthodes issues de la deuxième approche à la solution de référence.

Dans la simulation numérique testant ces méthodes, on se place dans un cas test simple. Il s'agit d'une particule se déplaçant dans un domaine rectangulaire de 0.1×0.1 sous l'effet d'un champ extérieur et des champs auto-consistants (B,E) engendrés par son déplacement. Au temps t=0, la particule est située au centre du domaine d'observation avec une vitesse initiale parallèle à l'un des axes du domaine défini à partir de la norme du coefficient relativiste $\gamma^{-1} = \sqrt{1 - \frac{v^2}{c^2}}$. On définit une discrétisation structurée régulière du domaine avec respectivement n_x et n_y points le long de l'axe des abscisses et des ordonnées. L'implémentation de ces tests s'est faite à partir du code objet du modèle discrétisé présenté dans le chapitre précédent, par l'ajout de classes dérivées contenant les méthodes de correction et leurs données spécifiques. Dans nos tests, nous avons travaillé sur des maillages 40x40 et 80x80 en choisissant un pas de temps satisfaisant la condition de stabilité de notre modèle numérique ($c\,\Delta t \leq \frac{\Delta x}{\sqrt{2}}$). Nous avons pris un coefficient relativiste $\gamma = 650$ et appliqué un champ extérieur de 58 Tesla (Wb/m^2). Pour observer l'effet des champs consistants sur le mouvement de la particule (négligeable lorsque la charge est faible) nous appliquons à la particule élémentaire (électron) un poids de 1.e+15.

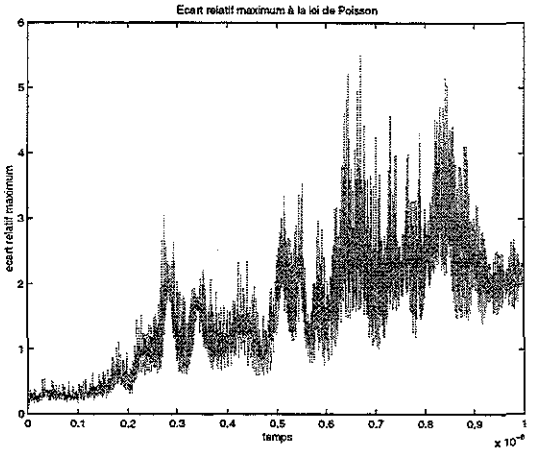


FIG. 5.1 – *Evolution de l'écart relatif maximum à la loi de Poisson dans le cas où aucune correction du champ électrique n'est effectuée et les densités de charge et de courant sont calculées par interpolation sur les noeuds du maillage.*

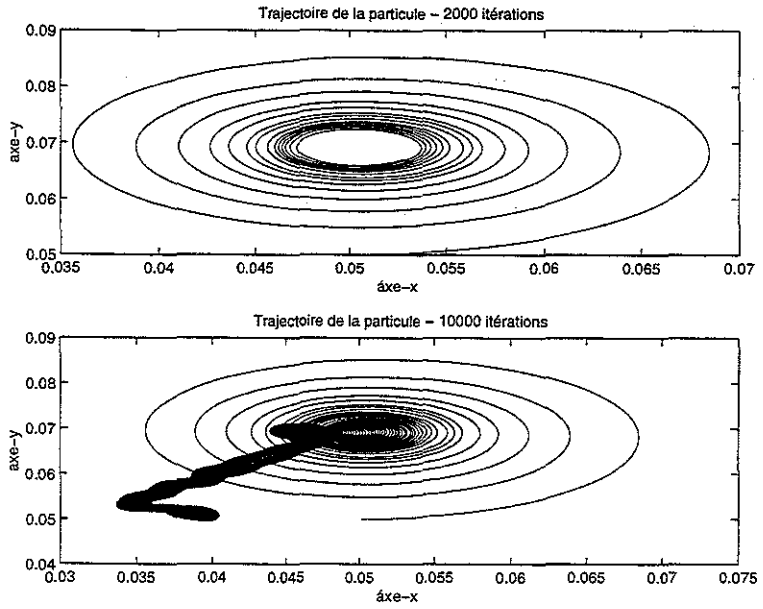


FIG. 5.2 – *Trajectoire de la particule sur 2000 itérations et 10000 itérations*

5.2 Méthodes de correction

5.2.1 Introduction

On considère les équations de Maxwell, j et ρ étant donnés :

$$\frac{\partial E}{\partial t} - c^2 \nabla \times B = -\frac{j}{\epsilon_0}, \tag{5.1}$$

$$\frac{\partial B}{\partial t} + \nabla \times E = 0, \tag{5.2}$$

$$\nabla . E = \frac{\rho}{\epsilon_0}, \tag{5.3}$$

$$\nabla . B = 0 \tag{5.4}$$

où E, B, ρ et j désignent respectivement le champ électrique, le champ magnétique d'induction, la densité de charge et la densité de courant. On considère un domaine borné Ω dans un milieu homogène où la permittivité diélectrique et la perméabilité magnétique sont constantes. L'équation de conservation de la charge s'écrit

$$\frac{\partial \rho}{\partial t} + \nabla . j = 0. \tag{5.5}$$

Si ρ et j satisfont (5.5), E_0,B_0 les conditions initiales sur les champs électromagnétiques vérifient (5.3)-(5.4). Alors (5.1)-(5.4) ont une unique solution dans une espace correctement choisi. Avec ces hypothèses, si on calcule la solution à partir de (5.1) et (5.2), on a un système bien posé et (5.3) et (5.4) sont automatiquement satisfaites. Pour le montrer, il suffit de prendre la divergence des équations (5.1) et (5.2).

Cependant, dans un code particulaire où ρ et j sont obtenus sur les noeuds du maillage permettant de résoudre les équations de Maxwell, par interpolation des informations provenant des particules, l'équation de conservation de la charge (5.5) n'est pas satisfaite de façon exacte et (5.1)-(5.4) est *surdéterminé*. Même si l'écart à (5.5), dû aux approximations successives, est petit à chaque itération en temps, elle s'accumule sur des longs pas de temps et l'écart à la loi de Poisson (5.3) peut devenir arbitrairement grand. En effet, soit "**a**" l'écart à (5.5), $a = \nabla.j + \frac{\partial \rho}{\partial t}$, en prenant la divergence de (5.1), on obtient :

$$\frac{\partial}{\partial t}(\nabla.E - \frac{\rho}{\epsilon_0}) = -\frac{a}{\epsilon_0}.$$

L'équation de Poisson est vérifiée au temps t=0, on a donc

$$\nabla.E - \frac{\rho}{\epsilon_0} = -\frac{at}{\epsilon_0}.$$

Même si "**a**" est petit à chaque itération en temps, sur des temps longs, l'écart à la loi de Poisson peut devenir arbitrairement grand. Un second type d'erreur peut survenir, dû au fait que la divergence du rotationnel ne soit pas nulle dans le modèle discret utilisé, ce n'est pas le cas pour le schéma numérique choisi (schéma de Yee).

Les figures (5.1, 5.2) montrent respectivement l'erreur relative sur l'équation de Poisson et la trajectoire de la particule dans le cas où ρ et J sont calculées par interpolation sur les noeuds du maillage et aucune correction n'est effectuée sur le champ électrique après la résolution des équations de Maxwell.

Dans les méthodes qui ont été étudiées jusqu'à présent, on distingue deux approches :

- La première approche est l'utilisation d'un algorithme plus compliqué pour calculer les densités de courant et de charge sur les noeuds du maillage, de telle façon que l'équation de conservation de la charge (5.5) soit exactement vérifiée ([79], [50]). Nous prendrons les solutions issues de cette approche comme référence dans la comparaison des différentes méthodes sur un cas test.
- La deuxième approche consiste à trouver une formulation annexe qui soit bien posée ([52]) et qui évite les accumulations d'erreur sur la divergence du champ électrique. Cette approche a déja été formulée dans ([3]) en introduisant un potentiel correcteur comme multiplicateur de Lagrange dans l'équation d'Ampère. Elle est également généralisée dans ([53]) en associant à un opérateur différentiel donné g, une formulation annexe :

$$\frac{\partial E}{\partial t} - c^2 \nabla \times B + c^2 \nabla p = -\frac{j}{\epsilon_0}, \tag{5.6}$$

$$\frac{\partial B}{\partial t} + \nabla \times E = 0, \tag{5.7}$$

$$g(p) + \nabla.E = \frac{\rho}{\epsilon_0}, \tag{5.8}$$

$$\nabla.B = 0. \tag{5.9}$$

Cette nouvelle variable "p" ajoute un nouveau degré de liberté dans les équations de Maxwell, elle sera couplée à la condition sur la divergence de E (5.8) par l'opérateur g(p). En appliquant l'opérateur de divergence à (5.6) et en dérivant (5.8), on montre que "p" vérifie :

$$\frac{\partial g(p)}{\partial t} - c^2 \Delta p = \frac{1}{\epsilon_0}(\frac{\partial \rho}{\partial t} + \nabla . j). \tag{5.10}$$

Pour j, et ρ donnés qui ne satisfont pas nécessairement les équations de conservation de la charge, p solution de (5.10), il est montré dans ([52]) que pour des conditions initiales et des conditions frontières appropriées et sous certaines hypothèses classiques de régularité, le système associé à la formulation annexe admet une solution unique. Dans la suite de ce chapitre, nous indiquerons pour chaque opérateur différentiel g traité, les conditions limites et les conditions frontieres utilisées.

On retrouve dans cette approche les méthodes généralement utilisées dans un code PIC, les méthodes de Boris ([11]) et de Marder-Langdon ([47]), ([43]), en considérant successivement les cas où :

- g(p)=0 : p est solution d'une équation de type Poisson, on obtient une formulation *hyperbolique-elliptique* de Maxwell. Cette approche est équivalente à la correction de Boris.
- $g(p) = \frac{p}{d}$: p est solution d'une équation de type chaleur ce qui donne une formulation *hyperbolique-parabolique* de Maxwell. Cette approche est équivalente à la correction de Marder-Langdon.
- $g(p) = \frac{1}{\chi^2}\frac{\partial p}{\partial t}$: p est solution d'une équation de type ondes, on obtient une formulation purement *hyperbolique* de Maxwell.

Dans ce qui suit, on montrera que pour ces différentes méthodes, l'écart à l'équation de Poisson (5.3) est plus ou moins faible, elle n'augmente pas avec le temps. On comparera les solutions obtenues avec la solution issue de la première approche, choisie comme référence .

5.2.2 Méthodes respectant l'équation de conservation de la charge

Dans ce paragraphe, nous décrivons une méthode de calcul des champs électromagnétiques dans un code PIC en dimension 2 qui permet de déterminer directement les solutions du problème (5.1-5.4) à partir d'un calcul précis des densités de charge et de courant sur les noeuds du maillage en évitant l'utilisation des méthodes de correction du champ électrique, qui sont couteuse en temps CPU et qui peuvent fournir des solutions qui s'éloignent du modèle physique. Elle s'applique également en dimension 3 (voir [79]). Il s'agit d'une méthode de calcul de la densité de courant sur un maillage structuré vérifiant de *façon exacte* l'équation de conservation de la charge (5.5) à partir de l'information du déplacement des charges. Le champ électrique E obtenu à partir des équations (5.1) et

(5.2) par un schéma d'ordre deux en temps et en espace, satisfait alors automatiquement l'équation de Poisson (voir figure 5.3).

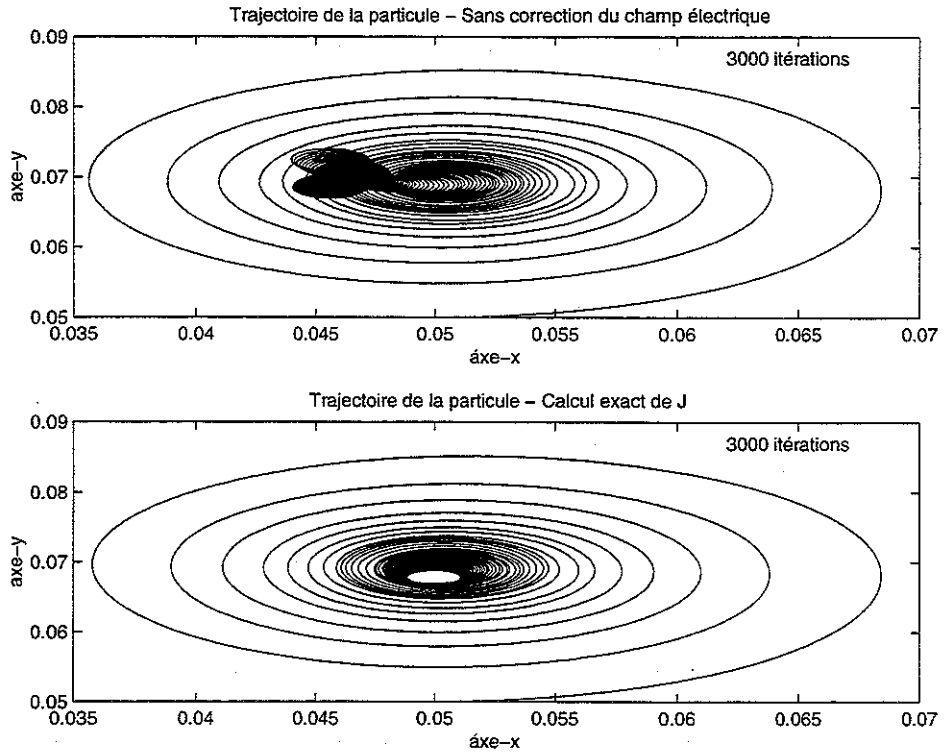


FIG. 5.3 – *Trajectoire de la particule*

Calcul de la densité de courant

Contrairement à une mise à jour séparée des parties longitudinale et transverse du champ électrique qui est souvent effectuée par les solveurs utilisant les transformées de Fourier, la mise à jour est globale à partir des équations de Maxwell. On peut faire les observations suivantes :

- A chaque pas de temps, le champ magnétique B modifie uniquement la partie transverse du champ électrique (5.1),
- on récupère la modification de la partie longitudinale de E, ie son flux, dans la partie longitudinale de J, qui, en vertu de l'équation de conservation de la charge (5.5) correspond à une modification de la densité de charge sur un pas de temps,
- à chaque pas de temps, le champ magnétique B est mis à jour par la partie transverse du champ électrique E.

En d'autres termes, il n'est pas nécessaire de calculer la partie longitudinale de E séparément à chaque pas de temps, sauf au début des itérations en temps. Cela suppose que l'expression en différences finies du courant est consistante avec l'équation de conservation

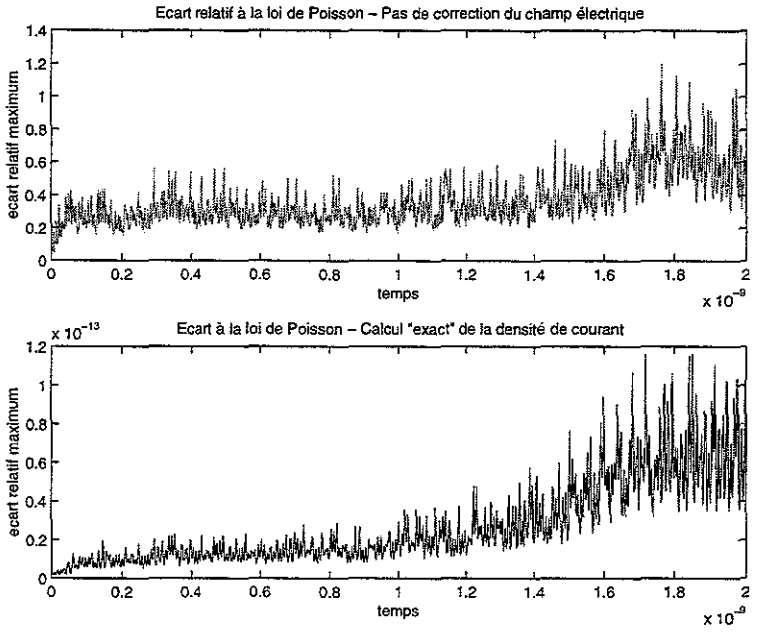


FIG. 5.4 – *Erreur relative maximum sur l'équation de Poisson (ordre de grandeur différent en ordonnées)*

de la charge, le flux J représentant rigoureusement la modification en temps de la densité de charge. A chaque itération en temps de la simulation, chaque charge est amenée à se déplacer d'une certaine distance, dans une certaine direction. Dans le cas discret, la densité de courant est représentée par le mouvement des particules.

La méthode est décrite dans ([79]). On crée une grille de carrés de taille 1 et à chaque particule ou charge, arbitrairement placée sur la grille, on associe un carré de taille 1 centré sur cette particule ou cette charge. On suppose que la quantité de charge est uniformément répartie sur la surface du carré et chaque charge contribue à la densité de charge des quatre mailles voisines proportionnellement à la surface du rectangle, intersection du carré de la charge et de la maille (voir Fig :5.5). C'est la méthode "area weighting" ([11]).

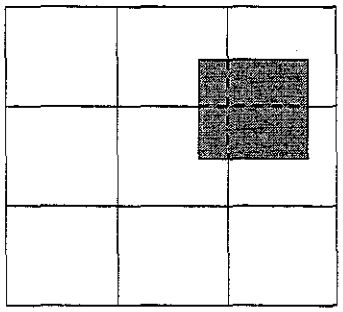


FIG. 5.5 – *Carré de charge associé à chaque particule se déplaçant dans un maillage structuré*

On peut remplacer chaque frontière d'une maille par une partie du carré de charge qui

correspond à la variation de charge sur un pas de temps dans une direction donnée (figure:5.6). Le déplacement d'une particule modifie le courant à travers quatre frontières dans la plupart des cas mais on devra tenir compte de la modification du courant à travers sept ou dix frontieres (figures 5.7 et 5.8).
Le pas de temps doit satisfaire la condition de courant $v\Delta t < \frac{\Delta x}{\sqrt{2}}$ en dimension 2. Ce qui

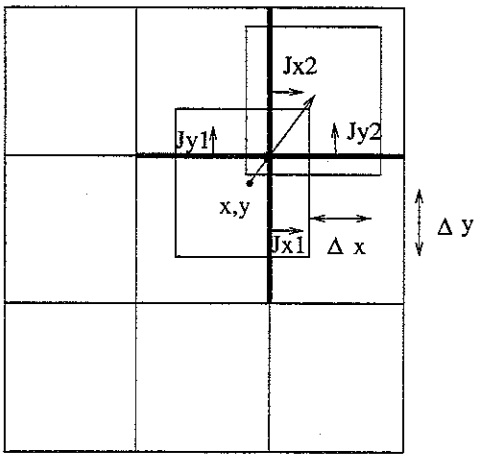


FIG. 5.6 – *Cas le plus fréquent où le déplacement de la charge modifie le courant à travers 4 frontières*

limite le déplacement à moins de $\frac{\Delta x}{\sqrt{2}}$ par pas de temps.

Dans le cas ou quatre frontières sont concernées, le déplacement des charges à travers les frontières donnent naissance à des courants J_{x1},J_{x2},J_{y1},J_{y2} comme indiqué dans la figure (5.6). Si on définit "l'*origine de la charge*", comme l'intersection des frontières des mailles les plus proches du centre de la charge au départ du mouvement, x et y, les coordonnées du centre de la charge par rapport à ce point, on peut écrire l'expression du courant à travers les frontières des mailles en calculant la quantité de charge traversant ces frontières. On a:

$$J_{x1} = q\Delta x(\frac{1}{2} - y - \frac{1}{2}\Delta y) \tag{5.11}$$

$$J_{x2} = q\Delta x(\frac{1}{2} + y + \frac{1}{2}\Delta y) \tag{5.12}$$

$$J_{y1} = q\Delta y(\frac{1}{2} - x - \frac{1}{2}\Delta x) \tag{5.13}$$

$$J_{y2} = q\Delta y(\frac{1}{2} + x + \frac{1}{2}\Delta x) \tag{5.14}$$

où q est la charge de la particule. Dans le cas d'un déplacement sur quatre frontières, la quantité de calcul nécessaire n'est pas plus importante que pour la méthode qui permet la mise à jour de la partie longitudinale de E. Les formules utilisées (5.11-5.14) sont les mêmes que celles utilisées par Morse et Nielsen ([51],[11]). Mais en général, le déplacement

des charges peut affecter le courant sur plus de quatre frontières. Le cas de sept frontières est décrit dans la figure (5.7). Il sera traité comme la décomposition de deux déplacements sur quatre frontières successifs (voir [79]).
Dans des cas plus exceptionnels, le déplacement des charges peut modifier le courant sur 10 frontières (voir figure 5.8) et comme pour les déplacements sur 7 frontières, un déplacement sur 10 frontières sera décomposé en 3 déplacements sur 4 frontières successifs (voir [79]).

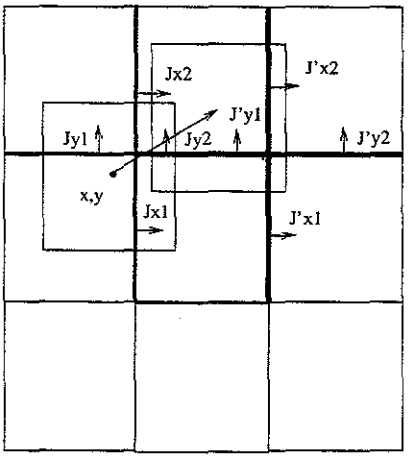


FIG. 5.7 – *Cas où le déplacement de la charge modifie le courant à travers 7 frontières*

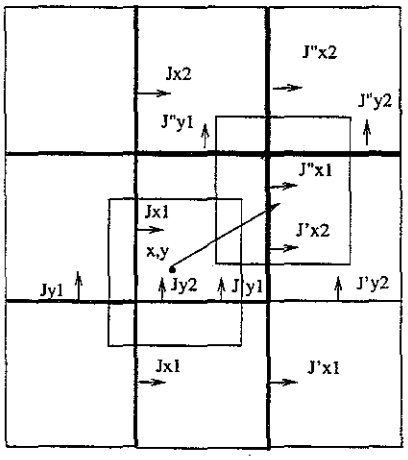


FIG. 5.8 – *Cas où le déplacement de la charge modifie le courant à travers 10 frontières*

La solution du problème (5.1-5.2) obtenue en utilisant cette méthode de calcul de la densité de courant sur les noeuds du maillage est numériquement solution de l'équation de Poisson (voir figure 5.4), on a une solution directe du problème (5.1)-(5.4). *On prendra cette solution comme référence pour analyser le comportement de la solution des équations de Maxwell lorsqu'on améliore l'erreur sur la divergence par une méthode de correction issue de la deuxième approche.* La solution obtenue sans correction du champ électrique

et par interpolation des densités de courant sur les noeuds du maillage s'écarte assez rapidement de cette solution de référence (figure 5.9). Mathématiquement pour que la solution de (5.1-5.2) soit solution de (5.3) et (5.4), il est nécessaire que E et B satisfassent ces équations au temps t=0, cette méthode est très sensible aux conditions initiales. Dans notre implémentation, nous effectuons une correction de Poisson pour déterminer E_0.

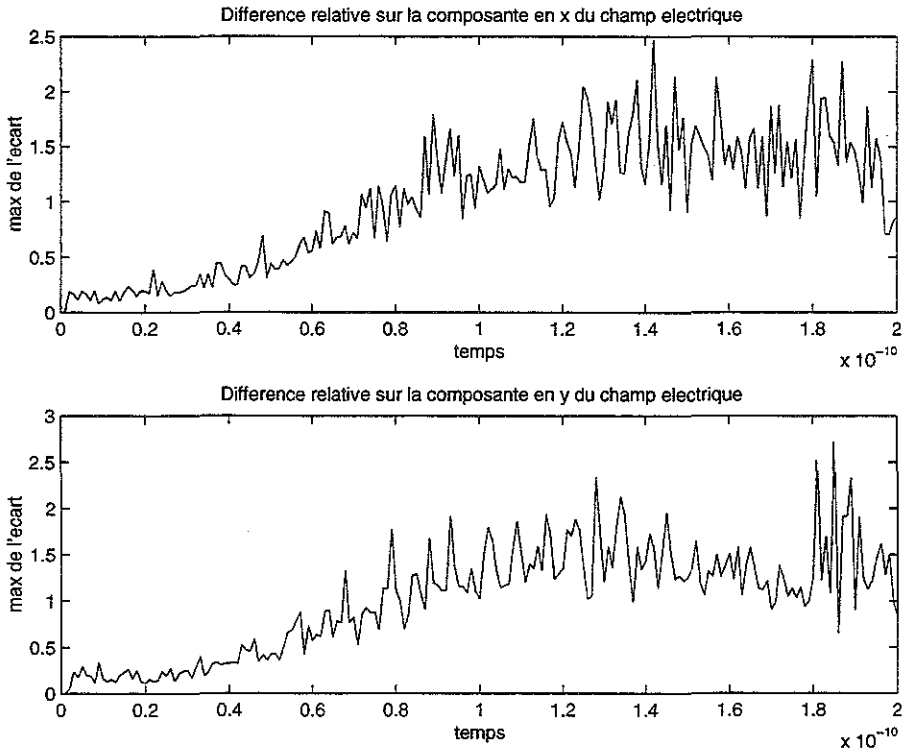


FIG. 5.9 – *Ecart relatif maximum de la solution sans correction à la solution de référence*

5.2.3 Méthode de Boris

Algorithme

La méthode de Boris correspond au cas où le potentiel correcteur est solution d'une équation de Poisson (g(p)=0), elle consiste à mettre à jour le champ électrique E en utilisant J non corrigé dans l'équation d'Ampère, puis en ajustant la partie longitudinale de E pour corriger la divergence de E ([11]).

$$E_{corrige} = E - \nabla\delta\phi$$

tel que :

$$\nabla\epsilon_0 E_{corrige} = \rho$$

où $\delta\phi$ est le potentiel correctif. $E_{corrige}$ satisfait la loi de Poisson :

$$\nabla.(\epsilon_0 E - \epsilon_0\nabla\delta\phi) = \rho.$$

Cela revient à résoudre l'équation elliptique (5.15) avec E et ρ donnés :

$$\nabla \epsilon_0 \nabla \delta\phi = \nabla . \epsilon_0 E - \rho. \tag{5.15}$$

E et B satisfont les conditions limites imposées sur les champs, si on impose au potentiel correctif $\delta\phi$ une condition limite de Dirichlet homogène nulle sur la frontière du domaine d'observation, $E_{corrige}$ satisfait également ces conditions limites. *La correction de Boris n'agit que sur la partie irrotationnelle du champ électrique.* C'est une propriété importante car, c'est la partie du champ électrique qui n'intervient pas dans la mise à jour du champ magnétique. La propagation des ondes électromagnétiques est conservée, seules les composantes électriques sont affectées.

Discrétisation

La méthode de Boris est implémentée dans notre code en utilisant la discrétisation classique d'ordre 2 à 5 points du Laplacien. E_x étant connu sur les points $(i+\frac{1}{2},j)$ du maillage, E_y sur les points $(i,j+\frac{1}{2})$ du maillage, ρ et ϕ sur les points (i,j). On résout

$$(\Delta\phi)_{i,j} = (\frac{\partial E_x}{\partial x} + \frac{\partial E_y}{\partial y})_{i,j} - \rho_{i,j}$$

par le schéma :

$$\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{\Delta x^2} + \frac{\phi_{i,j+1} - 2\phi_{i,j} + \phi_{i,j-1}}{\Delta y^2} = \frac{E_{x,i+\frac{1}{2},j} - E_{x,i-\frac{1}{2},j}}{\Delta x} + \frac{E_{y,i,j+\frac{1}{2}} - E_{y,i,j-\frac{1}{2}}}{\Delta y} - \rho_{i,j}.$$

On est donc amené à résoudre un système de taille $(n_x \times n_y)^2$. La méthode de Boris fournit une correction globale de l'erreur sur la divergence de E. Cette erreur est par construction très faible (figure 5.11), elle correspond à l'ordre d'approximation du solveur de Poisson utilisé. Sur notre cas test, la figure (5.12) nous donne des indications sur l'évolution en temps de l'écart de la solution obtenue par cette méthode à la solution de référence. Si on observe l'évolution de cet écart dans le cas d'un calcul sans correction du champ électrique (figure 5.9), on voit que la correction de Boris maintient cet écart à sa valeur initiale alors qu'il croit progressivement dans le cas sans correction. On peut également observer ce phénomène sur les trajectoires de la particule (figures 5.3, 5.10).

La résolution du système a été effectuée en utilisant une méthode directe (algorithme de factorisation LU). La correction se fait à chaque pas de temps, elle est très couteuse en temps de calcul et en place mémoire. D'autre part, il faut indiquer également que les solveurs directs perdent en précision lorsque la taille du système grandit. D'autres méthodes de résolution du système sont utilisés dans les codes PIC afin d'accélérer cette phase de correction , en particulier, l'algorithme itératif DADI (Dynamic Alternating Direction Implicit,[36]) analysé dans ([48]). La correction peut également être effectuée après un certain nombre d'itérations en temps.

La correction de Poisson est intéressante pour des géométries simples et pour une implémentation utilisant des solveurs directs rapides comme les FFT. Par contre, elle devient couteuse pour des grilles non régulières et elle complique l'implémentation parallèle du code.

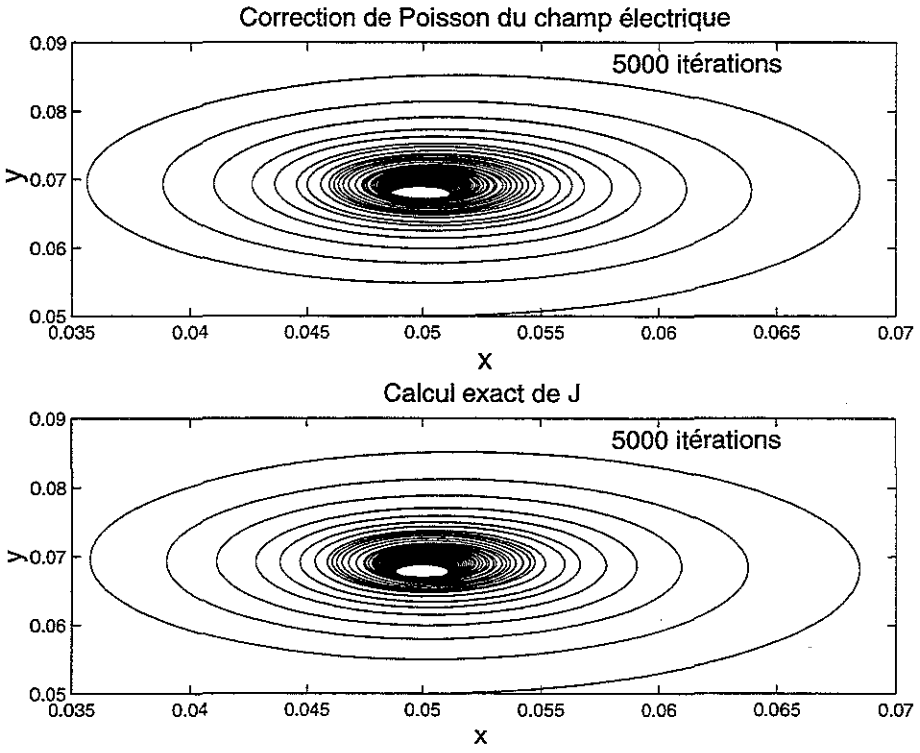


FIG. 5.10 – *Trajectoire de la particule*

5.2.4 Méthodes de Marder-Langdon

La méthode de Marder ([47]) correspond au cas ou le potentiel correcteur est solution d'une équation parabolique ($g(p) = \frac{p}{d}$), on a :

$$\begin{cases} \frac{\partial E}{\partial t} &= c^2 \nabla \times B - \frac{J}{\epsilon_0} + \nabla p, \\ p &= d\,(\nabla . E - \frac{\rho}{\epsilon_0}), \end{cases} \tag{5.16}$$

où ∇p est le *pseudo-courant* qui peut être considéré come un *pseudo-champ électrique* permettant de compenser l'erreur. Le nouvel opérateur choisi pour l'équation d'Ampère est consistant avec les équations de Maxwell.

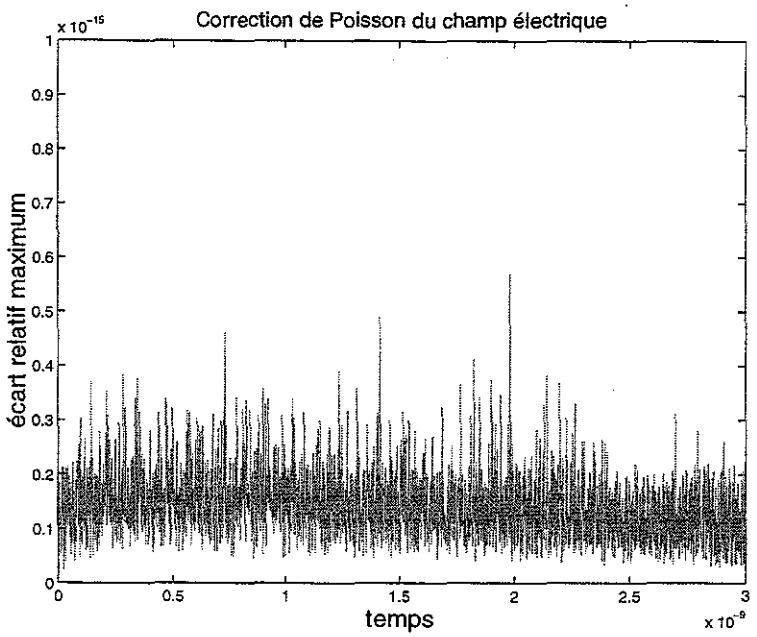


FIG. 5.11 – *Erreur relative maximum sur l'équation de Poisson*

Dans cette méthode la croissance de l'erreur est contrôlée par une procédure de diffusion ([47]), p vérifie l'équation de diffusion suivante :

$$\frac{\partial p}{\partial t} - d\ \Delta p = -\frac{d}{\epsilon_0}(\frac{\partial \rho}{\partial t} + \nabla.J). \tag{5.17}$$

Le terme source de (5.17), indique de quelle quantité s'écarte la densité de charge et la densité de courant du principe de conservation, un terme source non nul étant l'artefact de la méthode numérique utilisée pour calculer les densités de charge et de courant sur les noeuds du maillage.

Le paramétre "d" contrôle la vitesse de diffusion vers l'extérieur, il devra être choisi de telle façon que s'équilibre la vitesse de génération de p par le code due aux approximations et la vitesse de dissipation. L'algorithme doit être *auto-correcteur*. Il sera choisi suffisamment petit pour ne pas affecter la stabilité du système et suffisamment grand pour appliquer la fonction désirée (p=0). La restriction de stabilité introduite par cette addition est connue pour les équations de la chaleur en dimension 2 par :

$$d \leq \frac{1}{2\Delta t}\left(\frac{\Delta x^2\,\Delta y^2}{\Delta x^2 + \Delta y^2}\right) = dmax$$

où Δt et Δx sont respectivement les pas de discrétisation en temps et en espace.

Comme pour la correction de Poisson, on montre que la correction de Marder n'agit que sur la partie irrotationnelle du champ électrique. L'équation des ondes de B restent donc mathématiquement inchangée et numériquement puisque le rotationnel du gradient de Best nul dans le schéma utilisé. La propagation des ondes électromagnétiques est conservé, seules les composantes électriques sont affectées.

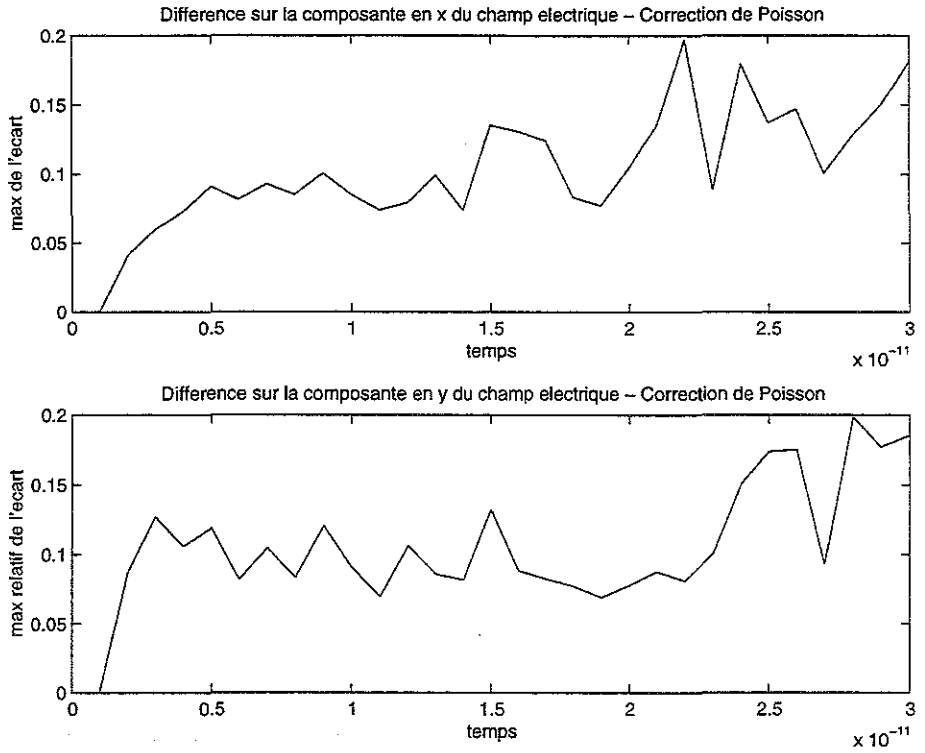


FIG. 5.12 – *Ecart relatif maximum de la solution avec correction de Poisson à la solution de référence*

Discrétisation

On utilise comme précédemment, le schéma en temps "leap-frog" ([11]) associé aux équations (5.16), on note respectivement $\tilde{\nabla}\times, \tilde{\nabla}., \tilde{\nabla}$ la version discrétisée des opérateurs $\nabla\times, \nabla., \nabla$, on a :

$$\begin{aligned} \frac{E^{n+1}-E^n}{\Delta t} &= c^2\,\tilde{\nabla}\times B^{n+\frac{1}{2}} - J^{n+\frac{1}{2}} + \tilde{\nabla}p^n, \\ p^n &= d\,(\tilde{\nabla}.E^n - \frac{\rho^n}{\epsilon_0}). \end{aligned} \tag{5.18}$$

Ce schéma centre correctement en temps les termes E et B mais pas le terme p. On obtient uniquement une précision d'ordre 1 pour ce terme. Langdon propose une correction du schéma qui fournira une meilleure précision sur l'équation de Poisson.

Si on prend la divergence de (5.18), on obtient :

$$\frac{p^{n+1}-p^n}{d\Delta t} - \tilde{\Delta}p^n = -(\frac{\rho^{n+1}-\rho^n}{\Delta t} + \tilde{\nabla}.J^{n+\frac{1}{2}}). \tag{5.19}$$

C'est un schéma explicite de discrétisation en temps de (5.17) qui nous donne des informations sur l'étendue de $d\Delta t$ propre à cette équation de diffusion.

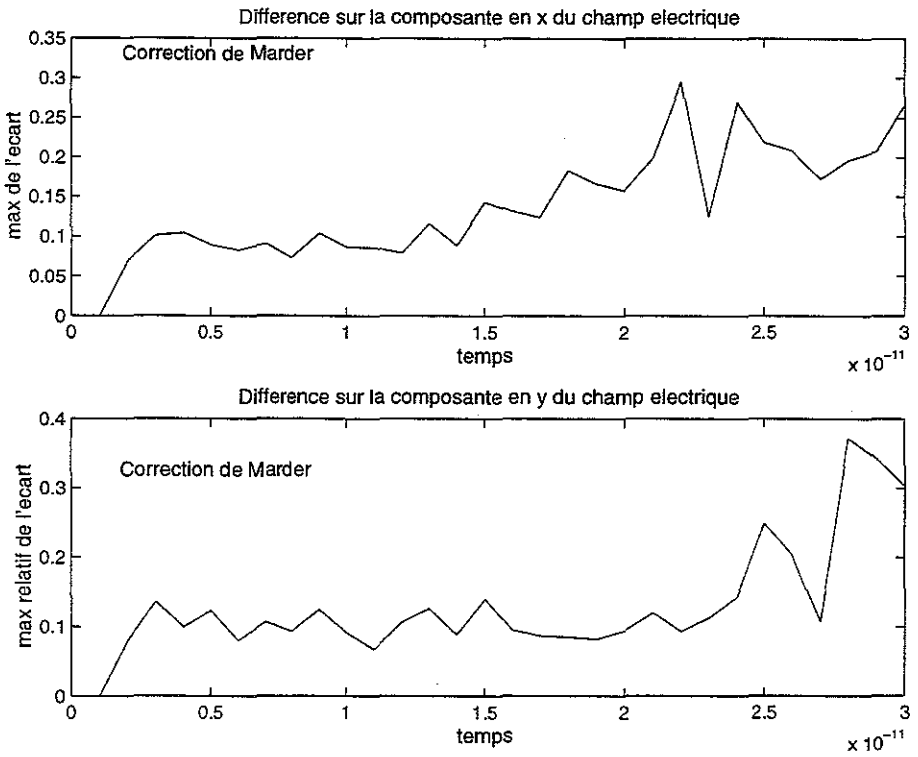


FIG. 5.13 – *Ecart relatif maximum de la solution avec correction de Marder à la solution de référence*

Modification de Langdon de la méthode de Marder

Les équations (5.18) utilisent les erreurs sur l'équation de Poisson au temps t^n, dues à l'erreur sur $\nabla.J^{n-\frac{1}{2}}$ et celles sur les temps précédents. Elle ne fait rien pour corriger les erreurs provenant $\nabla.J^{n+\frac{1}{2}}$. Cela peut être fait en prenant :

$$\begin{aligned} {}^{(0)}E^{n+1} - E^n &= c^2\Delta t\, \tilde{\nabla} \times B^{n+\frac{1}{2}} - \Delta t\, J^{n+\frac{1}{2}}, \\ {}^{(0)}p^{n+1} &= d\,(\tilde{\nabla}.{}^{(0)}E^{n+1} - \frac{\rho^{n+1}}{\epsilon_0}). \end{aligned} \tag{5.20}$$

On a :

$$E^{n+1} = {}^{(0)}E^{n+1} + \Delta t\, \tilde{\nabla}\,[d(\tilde{\nabla}.E^n - \frac{\rho^n)}{\epsilon_0})].$$

Cette expression est équivalente à la procédure de Marder. On tient compte de l'erreur provenant de $\nabla.J^{n+\frac{1}{2}}$ de façon plus précise en utilisant :

$$\begin{aligned} E^{n+1} &= {}^{(0)}E^{n+1} + \Delta t\, \tilde{\nabla}\, {}^{(0)}p^{n+1}, \\ &= {}^{(0)}E^{n+1} + \Delta t\, \tilde{\nabla}\,[d(\tilde{\nabla}.E^{n+1} - \rho^{n+1})]. \end{aligned} \tag{5.21}$$

Il est montré ([43]) que dans les deux schémas, le schéma de Marder et le schéma révisé de Langdon, p et ${}^{(0)}p$ évoluent comme la solution de (5.17). En effet, si on prend la

divergence de (5.20) et (5.21) on a:

$$\begin{aligned} \tilde{\nabla}.^{(0)}E^{n+1} &= \tilde{\nabla}.E^n - \Delta t\ \tilde{\nabla}\ .J^{n+\frac{1}{2}}, \\ \tilde{\nabla}.E^{n+1} &= \tilde{\nabla}.^{(0)}E^{n+1} + \Delta t\ \tilde{\nabla}^2\ .^{(0)}p^{n+1}. \end{aligned} \tag{5.22}$$

Ce qui nous donne en remplaçant n+1 par n dans la deuxième équation et en reportant $\nabla.\ E^n$ dans la première, que $^{(0)}p^n$ est solution de (5.17). On a donc $p^n =^{(0)} p^n$.
Si on compare les champs électriques obtenus à partir de ces deux corrections; on note E_1^n le champ obtenu au temps n par le schéma de Marder et E_2^n celui obtenu par le schéma révisé de Langdon, on a:

$$\begin{aligned} \tilde{\nabla}.\ E_1^n - \rho^n &= \frac{p^n}{d}, \\ \tilde{\nabla}.\ E_2^n - \rho^n &= \frac{p^n}{d} + \Delta t\ \tilde{\nabla}^2\ p^{n+1}. \end{aligned} \tag{5.23}$$

La seconde expression est plus petite pour des valeurs de d correspondant aux conditions

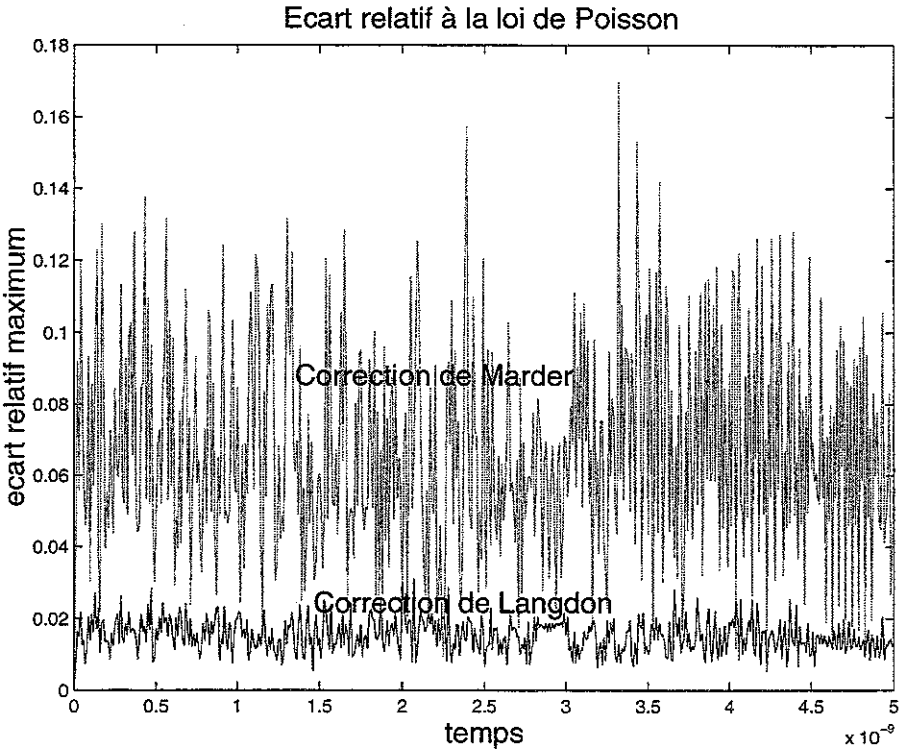


FIG. 5.14 – *Erreur relative maximum sur l'équation de Poisson pour la correction de Marder et la correction de Marder corrigée*

de stabilité de (5.19). Ce qui indique que pour des valeurs de d bien choisi, le schéma révisé de Langdon fournit une meilleure approximation de l'équation de Poisson (voir figure 5.14).

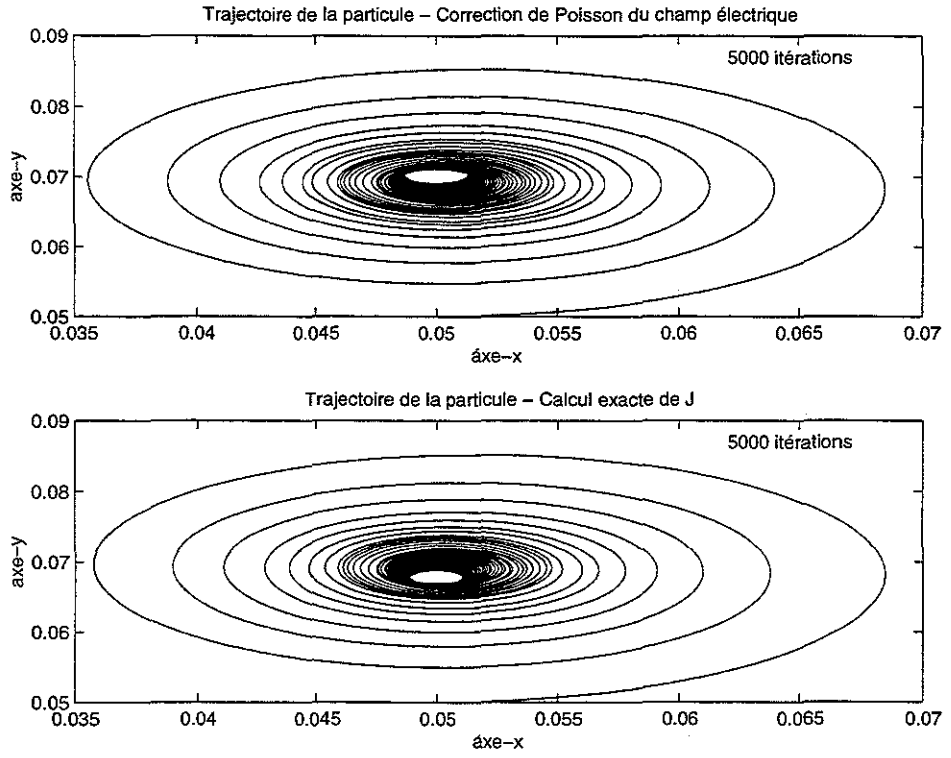


FIG. 5.15 – *Trajectoire de la particule*

Remarques

- L'équation (5.17) fournit des informations sur le mécanisme de correction. L'erreur locale de la conservation de la charge est diffusée par la fonction p. Si l'équation de conservation de la charge est satisfaite, le terme source de (5.17) est nul. Ce qui suggère les conditions initiales et limites suivantes :

$$\begin{array}{rcll} p(x,0) & = & 0 & \forall x \in \Omega, \\ p(x,t) & = & 0 & \forall x \in \partial\Omega \qquad \forall t \geq 0. \end{array} \tag{5.24}$$

- Sur notre cas test, on analyse conjointement le choix du paramètre "d" qui contrôle la vitesse de diffusion de l'erreur sur la divergence et l'écart de la solution obtenue à la solution de référence. On observe que si l'erreur sur l'équation de Poisson augmente plus on s'écarte de "dmax" (la vitesse de diffusion diminue), l'écart relatif maximum à la solution de référence évolue très peu (voir figures 5.17, 5.18, 5.19). Comme pour la correction de Boris, la correction de Marder-Langdon permet de stabiliser l'écart de la solution obtenue à la solution de référence.
- Langdon montre également que la méthode révisée de Marder est identique à la méthode de Boris dans laquelle la solution du système est calculée par une simple itération de l'algorithme de Jacobi. ([43]).
- Nielsen et Drobot montrent qu'un grand nombre d'itérations répétées de (5.21) convergent asymptotiquement vers la solution de la correction de Poisson ([56]). Dans la pratique quelques itérations suffisent pour améliorer la correction. On verra

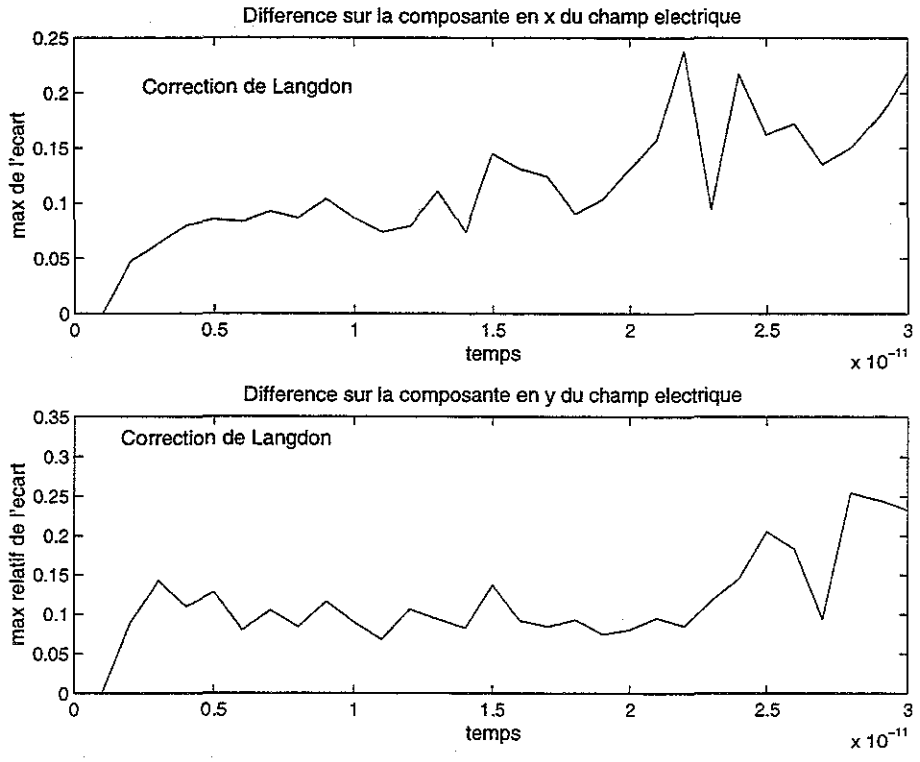


FIG. 5.16 – *Ecart relatif maximum de la solution avec correction de Langdon à la solution de référence*

sur notre cas test qu'il suffit de 5 itérations pour que l'erreur sur la divergence s'améliore et que l'écart de la solution obtenue à la solution de référence se stabilise (voir figures 5.20, 5.21, 5.22).

5.2.5 Méthode hyperbolique

L'approche "*purement hyperbolique*" est obtenue en choisissant dans (5.10) :

$$g(p) = \frac{1}{\chi^2} \times \frac{\partial p}{\partial t}$$

où le terme χ, adimensionalisé, représente la force de couplage entre l'équation d'Ampère (5.1) et l'équation de Poisson (5.3) .

L'équation d'évolution du potentiel p est donnée par l'équation des ondes :

$$\frac{\partial^2 p}{\partial t^2} - (\chi \times c)^2 \Delta p = \frac{\chi^2}{\epsilon_0}(\frac{\partial \rho}{\partial t} + \nabla . j) \tag{5.25}$$

équation aux dérivées partielles hyperboliques, admettant une solution et une seule sous certaines conditions initiales et aux limites, ayant la forme usuelle d'une onde se propageant à la vitesse (χc) vers l'extérieur. La taille de ce paramètre χ est inconnue et devra

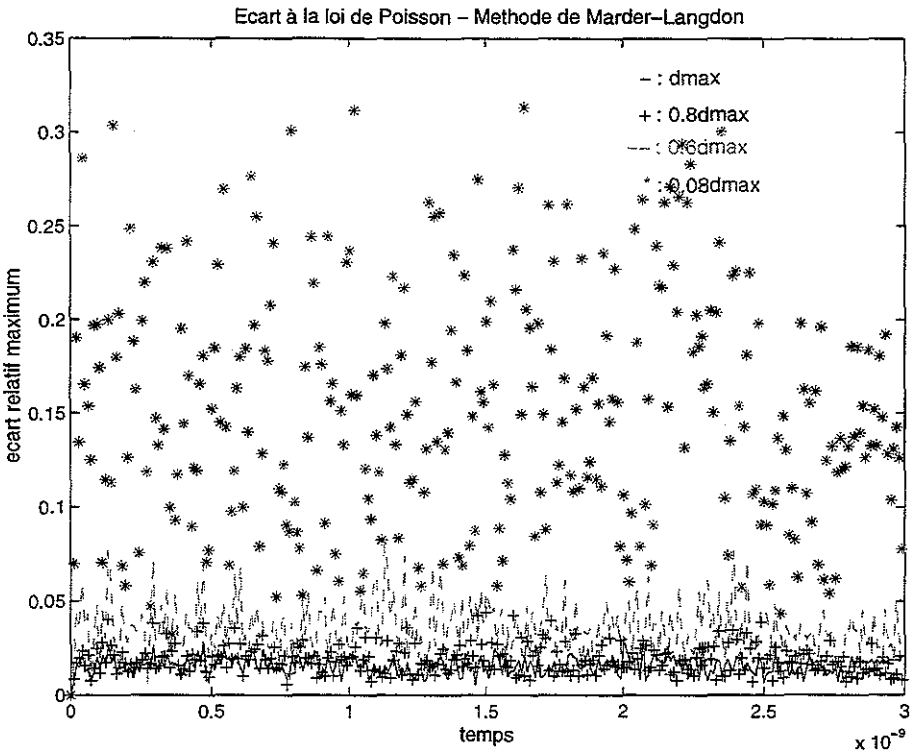


FIG. 5.17 – *Erreur relative maximum sur l'équation de Poisson avec différentes valeurs du paramètre de diffusion de la correction de Marder-Langdon*

être estimée par expérimentation numérique. Cependant, si l'on veut éliminer l'erreur sur E par **p**, aussi vite que la vitesse de propagation du champ électromagnétique, on prendra a priori $\chi \geq 1$.

Le terme source de (5.25) tend vers zéro si l'équation de conservation de la charge est satisfaite, on impose la condition initiale :

$$p(x,0) = 0 \quad \forall x \in \Omega.$$

Pour garantir que l'approche de cette nouvelle formulation satisfasse les équations de Maxwell approximativement à chaque pas de temps dans Ω, nous imposons des conditions d'ondes sortantes ([53]) (éliminant les ondes entrantes) qui sont du type $\mathrm{E.n} \pm (\chi\, c)\, p = 0$. Cette condition fournit la *stabilité* de la nouvelle formulation de l'équation d'Ampère dans Ω. D'autres conditions frontières peuvent être envisagées pour assurer la stabilité du système, elles sont examinées en particulier dans ([41]). Une implémentation de l'approche purement hyperbolique avec un schéma de volumes finis est décrite dans ([52]).

L'application de cette méthode de correction, se fait en deux temps : le calcul du potentiel "p" avec les conditions initiales et limites appropriées à partir de (5.8), puis la résolution de équations de Maxwell à partir de la formulation (5.6).

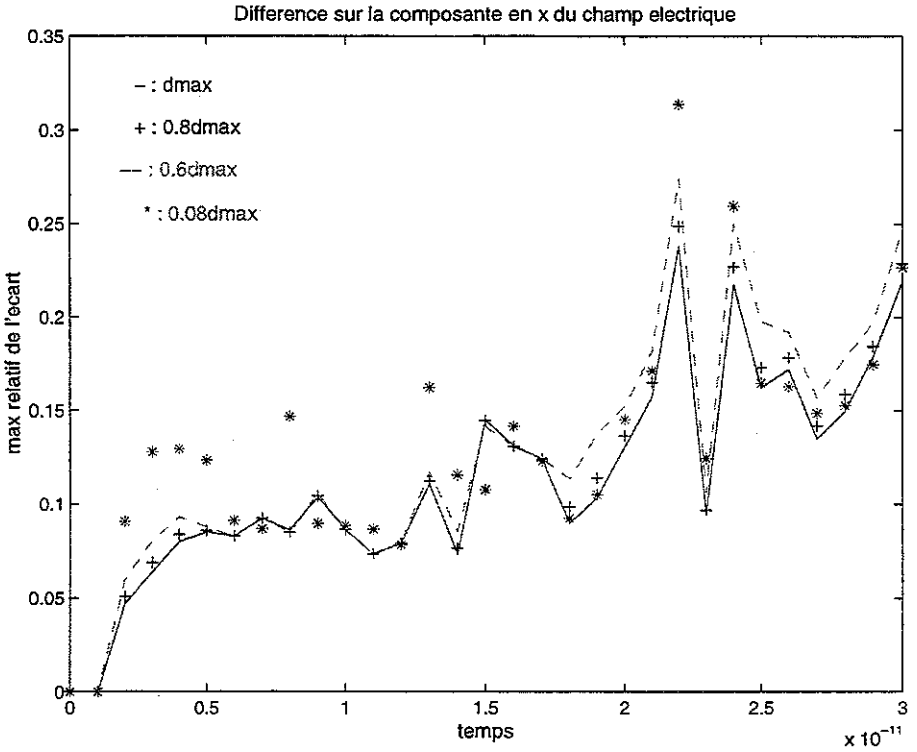


FIG. 5.18 – *Ecart relatif maximum de la composante en x du champ électrique issue de la correction de Langdon à la solution de référence avec différentes valeurs du paramètre de diffusion*

Discrétisation

On utilise le même schéma en temps "leap-frog" que pour les méthodes précédentes. On note au temps n+1, E^{n+1} une solution de (5.1-5.2) et $\tilde{E}^{n+1}$ une solution de (5.6-5.7), on a :

$$\begin{aligned}
\frac{E^{n+1} - E^n}{\Delta t} &= c^2\, \tilde{\nabla} \times B^{n+\frac{1}{2}} - J^{n+\frac{1}{2}}, \\
\frac{\tilde{E}^{n+1} - E^n}{\Delta t} &= c^2\, \tilde{\nabla} \times B^{n+\frac{1}{2}} - \Delta t\, J^{n+\frac{1}{2}} + c^2\, \tilde{\nabla} p^{n+\frac{1}{2}}, \qquad (5.26) \\
\frac{p^{n+\frac{1}{2}} - p^{n-\frac{1}{2}}}{\Delta t} &= \chi^2 (\tilde{\nabla}.E^n - \rho^n).
\end{aligned}$$

On montre facilement que :

$$\tilde{E}^{n+1} = E^n + c^2\, \Delta t\, \tilde{\nabla} p^{n+\frac{1}{2}}.$$

Comme pour le schéma de Marder, on centre correctement les termes E et B mais pas le terme "p". On obtient E et B avec une précision d'ordre 1 en temps et d'ordre 2 en espace. On pourra envisager, en utilisant le même principe que Langdon une modification de ce schéma qui permettra de tenir compte de l'erreur sur $\nabla.J^{n+\frac{1}{2}}$ au temps n+1.

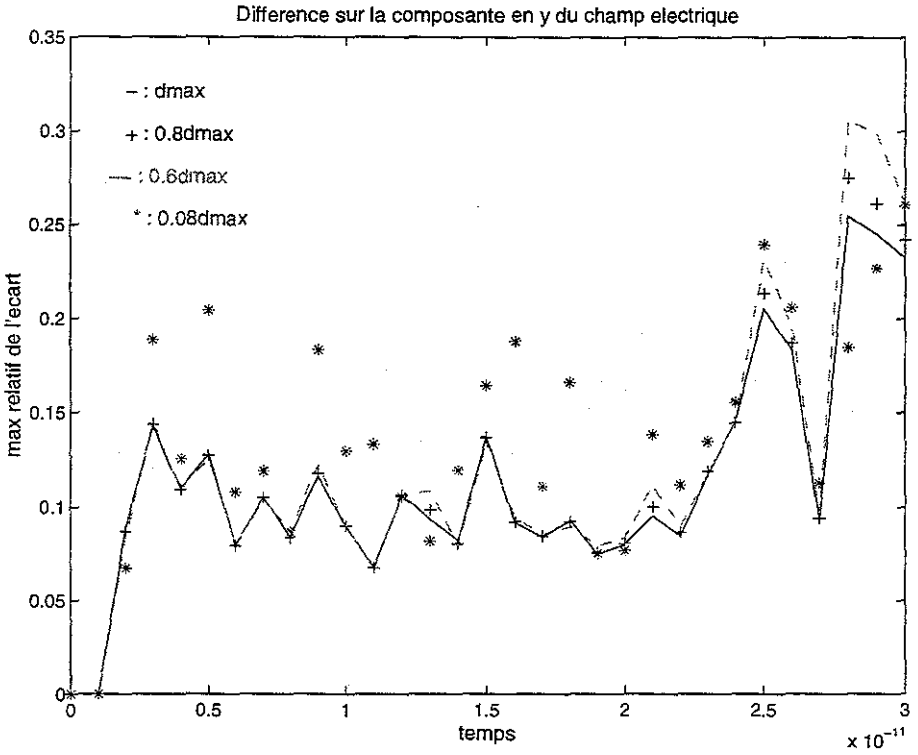


FIG. 5.19 – *Ecart relatif maximum de la composante en y du champ électrique issue de la correction de Langdon à la solution de référence avec différentes valeurs du paramètre de diffusion*

On a tenu compte des conditions limites imposées sur "p" en utilisant une méthode identique à l'implémentation des conditions de Silver Muller dans le solveur de Maxwell, en utilisant une discrétisation de l'équation (5.8) du problème annexe conjointement à la condition d'ondes sortantes.

5.3 Implémentation

Dans la description des méthodes de correction des champs électriques, nous pouvons observer que la correction se fait suivant deux approches :

- Le calcul se fait en deux étapes successives :
  - la résolution des équations d'Ampère et de Faraday (mise à jour de la partie transverse de E)
  - la correction des champs électriques obtenus par une méthode de type Boris, Marder, ...(mise à jour de la partie longitudinale de E)

  Ces deux étapes sont prises en charge par la méthode virtuelle "resolution" qui, après la première phase de calcul, fait appel à la méthode virtuelle "correctChamps".

  L'implémentation d'une nouvelle méthode de correction de ce type se fera par *dérivation* de la classe implémentant la résolution des équations de Maxwell. La classe

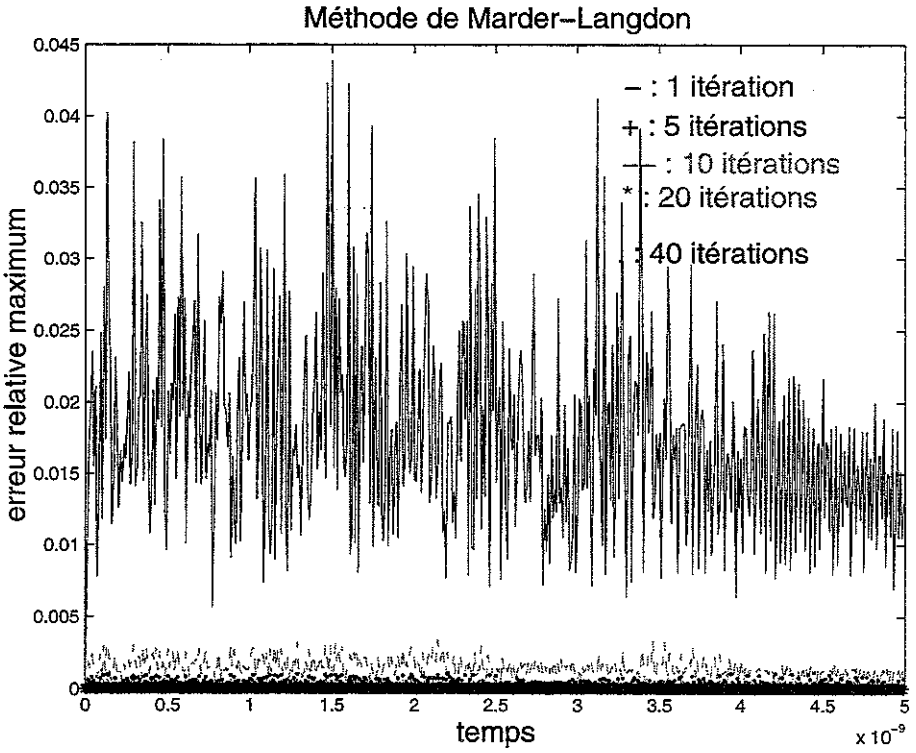


FIG. 5.20 – *Erreur relative maximum sur l'équation de Poisson avec différentes valeurs du nombre d'itérations effectuées*

dérivée contiendra les données spécifiques à la méthode de correction et une implémentation par la méthode virtuelle "correctChamps" de cet algorithme.

Dans notre application, nous avons implémenté :

- la méthode de correction de Boris par résolution directe de l'équation de Poisson dans la classe "**Champs2DBorisDirect**"
- la méthode de correction de Marder dans la classe "**Champs2DMarder**"
- la méthode de correction de Marder révisée par Langdon dans la classe "**Champs2DLangdon**"
- la méthode de correction purement hyperbolique dans la classe "**Champs2DHyp**"

La méthode "correctChamps" est définie *virtuelle pure* dans la classe "champ_discrets" qui, on l'a vu, est considérée comme une classe abstraite (aucun objet de ce type ne peut être instancié). Elle devra donc être définie par les classes dérivées ou déclarée à nouveau virtuelle pure. Elle a été implémentée dans notre application par une méthode vide dans la classe "champs2d". Elle implémente une résolution des équations de Maxwell qui utilise une interpolation linéaire des densités sur les noeuds du maillage et n'effectue aucune correction du champ électrique.

- Le calcul des densités de courant et de charge à partir des informations fournies par les particules se fait par *un algorithme respectant de manière exacte l'équation de conservation de la charge*. L'implémentation de cette approche se fera également

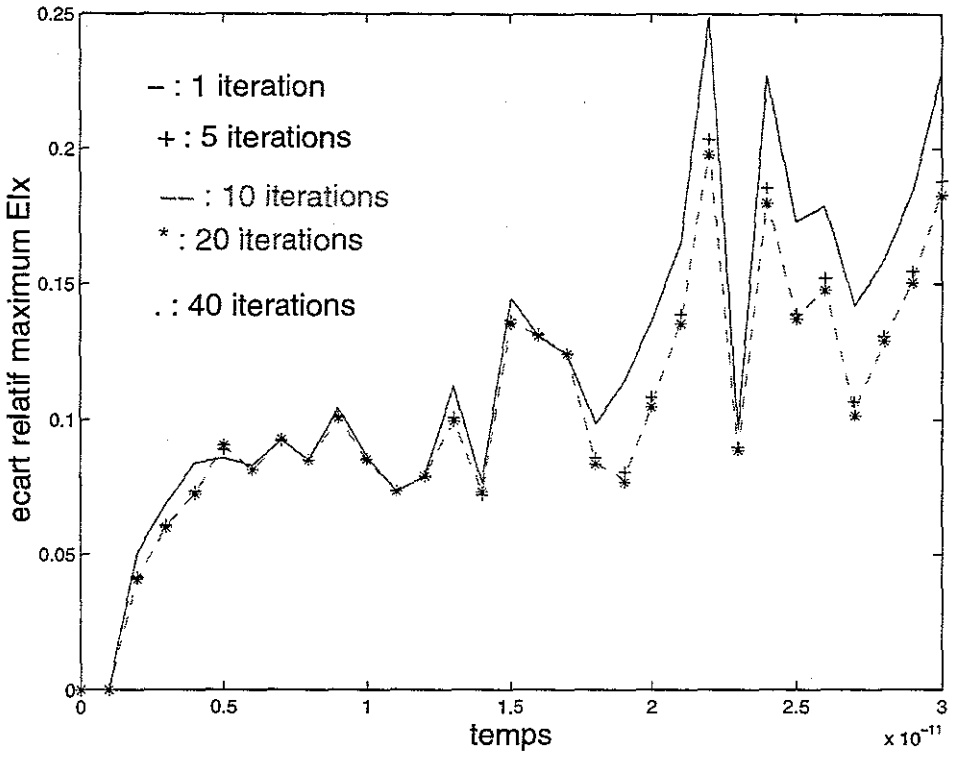


FIG. 5.21 – *Ecart relatif maximum de la composante en x du champ électrique issue de la correction de Langdon à la solution de référence avec différentes valeurs du nombre d'itérations effectuées*

par dérivation de la classe implémentant la résolution des équations de Maxwell. La classe dérivée
(**Champs2DConservCharge**) contiendra une implémentation de la méthode virtuelle "contribution" associée à l'algorithme de calcul des densités de courant décrite précédemment.

Pour la partie du code qui concerne la résolution des champs électromagnétiques, nous avons le schéma des classes décrit par la figure (Fig :5.26).

5.4 Conclusion

Le calcul de la densité de courant basé sur le déplacement des particules proposé par J. Villasenor et O. Buneman ([79]) nous fournit un algorithme direct et efficace de résolution numérique du problème (5.1-5.4). C'est donc une solution intéressante au problème d'erreur sur l'équation de Poisson due à l'approximation de la densité de courant sur les noeuds du maillage. Pourtant cet algorithme simple à mettre en oeuvre sur un maillage structuré devient complexe sur un maillage non structuré. Il est donc difficile à adapter à des méthodes de résolution de type éléments finis ou volumes finis et à des géométries

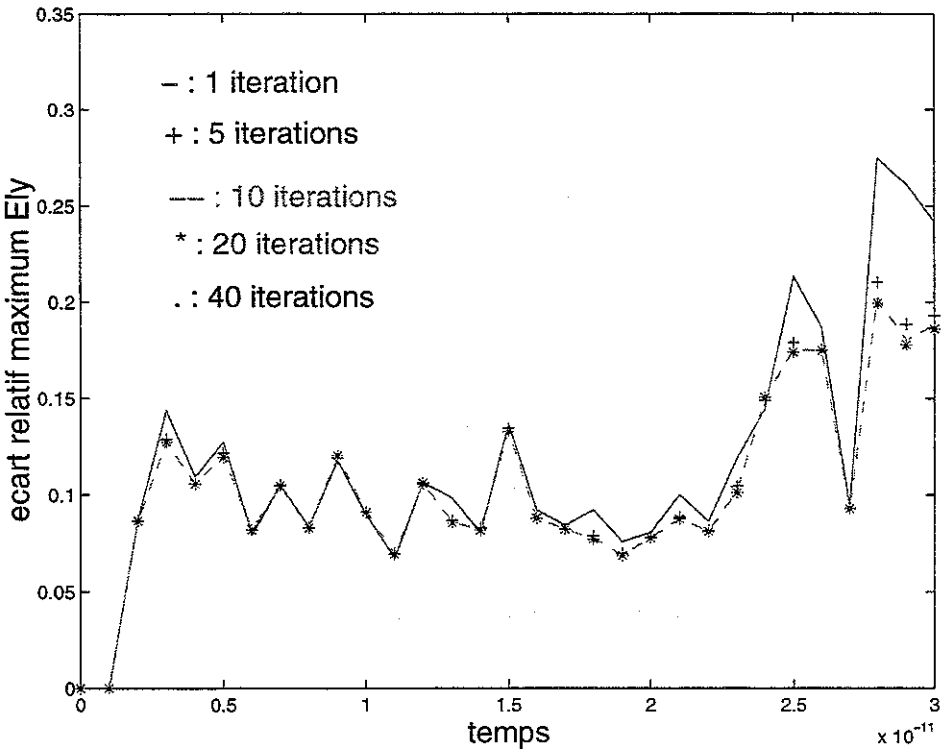


FIG. 5.22 – *Ecart relatif maximum de la composante en y du champ électrique issue de la correction de Langdon à la solution de référence avec différentes valeurs du nombre d'itérations effectuées*

complexes. Le coût en temps CPU de la complexité de l'algorithme n'a pu être analysé sur notre cas test puisque nous avons observé le comportement d'une seule macro-particule, ce coût est proportionnel au nombre de particules. Dans le cas d'un problème plus réaliste de physique des plasmas ce nombre est très important et le surcout de temps CPU ne sera pas si négligeable. Par contre le coût dû aux corrections issues de la deuxième approche est totalement indépendant du nombre de particules, il ne dépend que de la taille du maillage utilisé pour la résolution des équations de Maxwell. En ce qui concerne ces méthodes, la méthode de Boris fournit par définition une solution exacte de l'équation de Poisson contrairement aux méthodes de Marder, Langdon ou la méthode purement hyperbolique, mais elle devient très vite consommatrice de temps CPU et de mémoire puisqu'il s'agit de résoudre un système de taille $(n_x * n_y)^2$ à chaque itération en temps. La méthode de Boris est plus difficile à implémenter pour des géométries complexes, elle complique également l'implémentation parallèle du code. Ce qui est particulièrement évident en ce qui concerne notre code objet qui est conçu pour travailler de façon naturelle par décomposition de domaine par l'ajout de frontières sur les champs et les particules. Le modèle de discrétisation en temps explicite utilisé permet une parallélisation immédiate par décomposition de domaines qui se superposent sur les éléments frontières. L'ajout d'une méthode de correction de type Marder, Langdon ou purement hyperbolique ne change en rien la complexité de l'implémentation parallèle ce qui n'est pas le cas de la méthode de Boris.

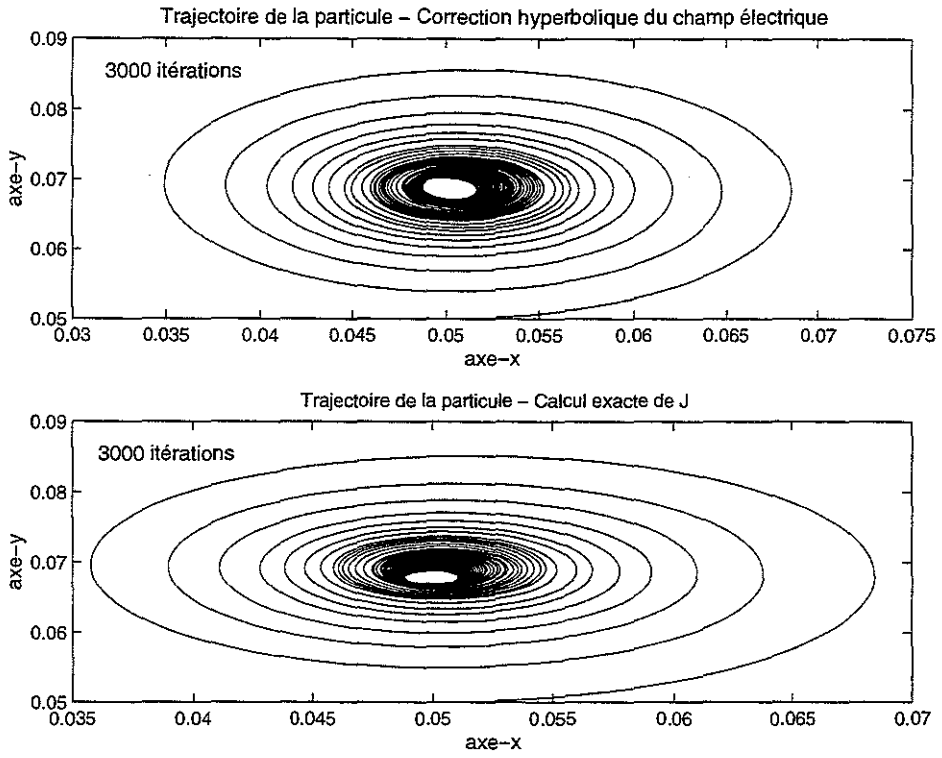


FIG. 5.23 – *Trajectoire de la particule*

L'analyse sur notre cas test de l'écart maximum relatif de la solution obtenue par ces différentes méthodes à la solution de référence nous indique qu'à partir d'un certain seuil d'erreur, cet écart se stabilise et ne s'améliore pas (voir figures 5.27, 5.21, 5.29). Sur ce cas test, il ne semble pas nécessaire d'effectuer une résolution exacte de l'équation de Poisson; quelques itérations de Marder-Landon (moins de 5) fournissent un résultat comparables sur le critère de l'écart relatif à la solution de référence. Sur ce cas test, ces méthodes permettent de stabiliser l'erreur sur la divergence ainsi que l'écart à la solution de référence; réduire l'erreur sur la divergence n'améliore pas le critère d'écart à la solution de référence. Il serait intéressant de faire des analyses identiques dans d'autres cas test en particulier pour des cas tests où le nombre de particules est important. En effet, la méthode de Boris calcule une correction globale de l'erreur sur la divergence contrairement aux trois autres méthodes de cette approche qui effectue des corrections locales donc efficaces sur notre cas test.

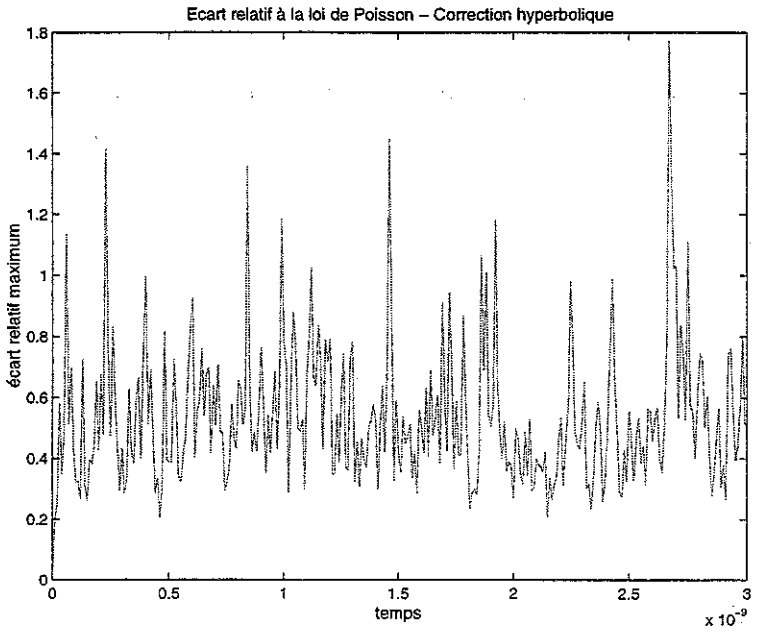


FIG. 5.24 – *Erreur relative maximum sur l'équation de Poisson*

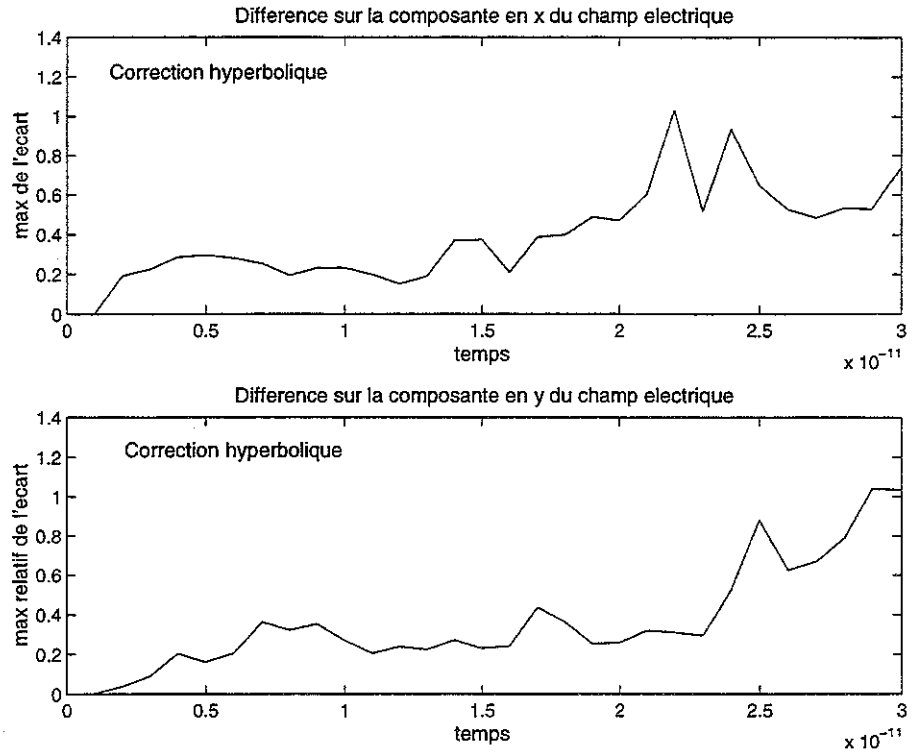


FIG. 5.25 – *Ecart relatif maximum de la solution avec correction hyperbolique à la solution de référence*

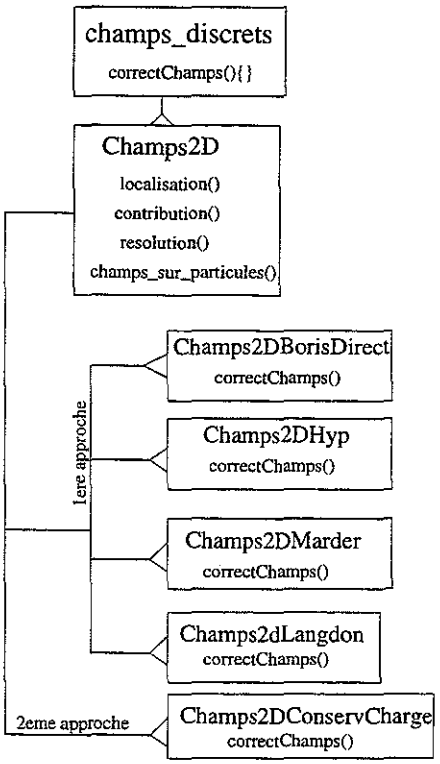


FIG. 5.26 – *Schéma des classes permettant la résolution des champs électromagnétiques*

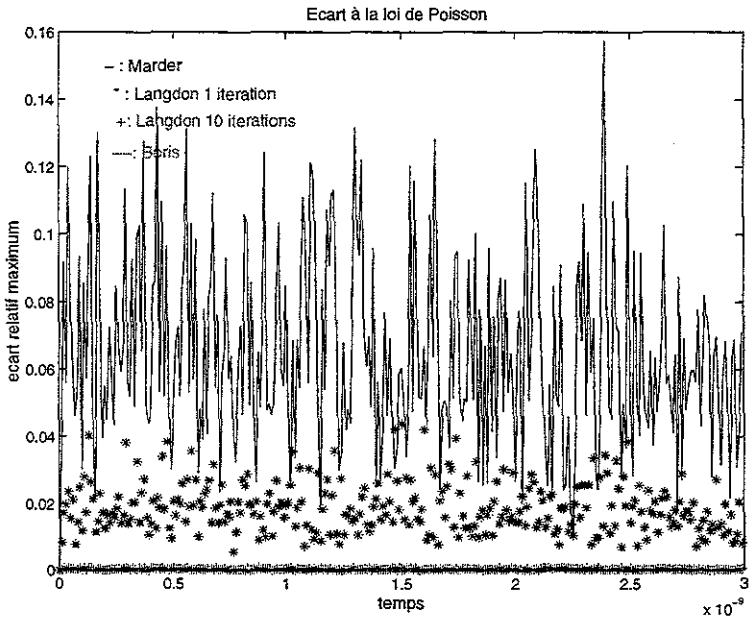


FIG. 5.27 – *Erreur relative maximum sur l'équation de Poisson pour différentes méthodes*

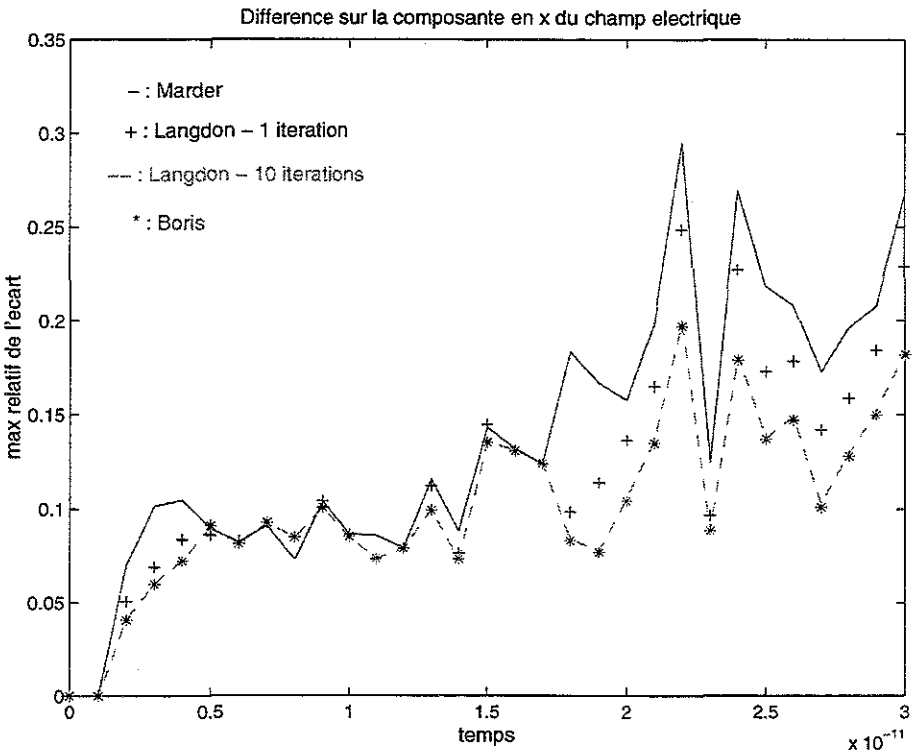


FIG. 5.28 – *Ecart relatif maximum de la composante en x du champ électrique à la solution de référence pour différentes méthodes*

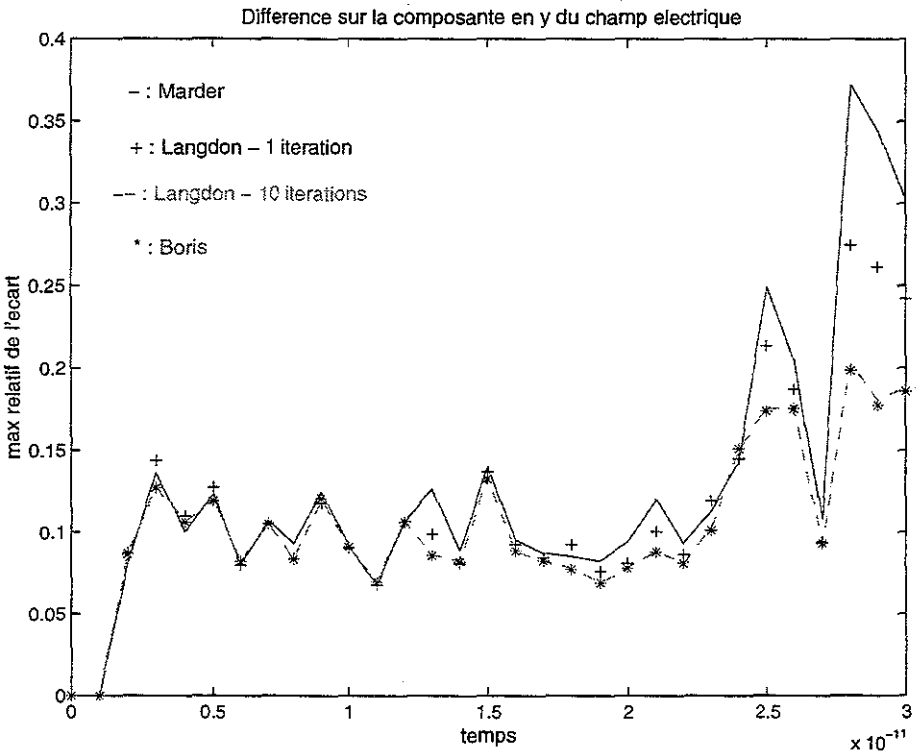


FIG. 5.29 – *Ecart relatif maximum de la composante en y du champ électrique à la solution de référence pour différentes méthodes*

Bibliographie

[1] A fast, free C FFT library home page. – 1998. http://theory.lcs.mit.edu/ fftw/homepage.html.

[2] Arakawa (A.). – Computational design for long-term numerical integration of the equation of fluid motions. *Journal of Computational in Physics*, vol. 1, 1993, pp. 119–143.

[3] Assous (F.), Degond (P.), Heintze (E.), Raviart (P.A.) et Segré (J.). – On a finite-element method for solving the three-dimensional Maxwell equations. *J.Comput.Phys.*, vol. 109, 1993, pp. 222–237.

[4] Balay (S.), Gropp (W.D.), McInnes (L.C.) et Smith (B.F.). – *PETSc 2.0 Users Manual.* – Rapport technique n ANL-95/11 - Revision 2.0.22, Mathematics and Computer Science Division - Argonne National Laboratory.

[5] Balay (S.), Gropp (W.D.), McInnes (L.C.) et Smith (B.F.). – PETSc home page. – 1998. http://www.mcs.anl.gov/petsc.

[6] Batchelor (G.K.). – *An introduction to fluid dynamics.* – Cambridge University Press, 1992, *Computational Physics*, volume 9.

[7] Benachour (S.), Roynette (B.) et Vallois (P.). – Branching process associated with 2d-Navier Stokes equation. *Revista Matematica Ibero Americana*, 1998.

[8] Bernadou (M.), George (P.L.) et Al. – *Une bibliothèque modulaire d' éléments finis.* – INRIA, 1988, Inria text book édition.

[9] Besson (O.), Bourgeois (J.), Chevalier (P.), Rappaz (J.) et Touzani (R.). – Numerical modelling of electromagnetic casting processes. *Journal of Computational Physics 92*, vol. 2, 1991, pp. 482–507.

[10] Besson (O.) et Rappaz (J.). – *Calculation of the Lorentz Forces in the Electromagnetic Casting.* – Rapport technique, AluSuisse,Chippis, 1986.

[11] Birdsall (C.) et Langdon (A.). – *Plasma physics via computer simulation.* – McGraw-Hill, New-York, 1985.

[12] Birman (K.), Cooper, R.and Joseph (T.), Kane (K.) et Schmuck (F.). – *The ISIS System Manual.* – Rapport technique nVersion1.2, 1989.

[13] Booch (G.). – *Object Oriented Design with Applications.* – Benjamin/Cummings Publishing Company,Inc., 1991.

[14] Buzzi-Ferraris (G.). – *Scientific C++: Building Numerical Libraries the Object-Oriented Way.* – Addison-Wesley, 1994.

[15] Cary (J. R.), Shasharina (S. G.), Cummings (J. C.), Reynders (J. V. W.) et Hinker (P. J.). – *Comparison of C++ and Fortran90 for Object-Oriented Scientific Pro-*

gramming. – Rapport technique nLA-UR-96-4064, Los Alamos National Laboratory, 1996.

[16] Coulaud (O.). – *Modélisation hautes fréquences en magnétohydrodynamique - Aspect magnétique.* – Rapport technique, INRIA, 1994.

[17] Coulaud (O.). – Multiple time scales and perturbation methods for high frequency electromagnetic-hydrodynamic coupling in the treatment of liquid metals. *Nonlinear Analysis, Theory, Methods et Applications*, vol. 30, n6, 1997, pp. 3637–3643.

[18] Coulaud (O.). – Asymptotic analysis of magnetic induction with high frequency for solid conductor. *Elsevier*, vol. 32, 1998, pp. 651–669.

[19] Cuvelier (C.). – *Some Numerical Methods for the Computation of Capillary free Boundaries Governed by Navier-Stokes Equations.* – Rapport technique n87-69, TU Delft, 1987.

[20] Decyk (V.K.), Norton (C.D.) et Szymansky (B.K.). – *Introduction to Object-Oriented Concepts using Fortran 90,.* – Rapport technique n PPG-1559, Institute of Plasma and Fusion Research Report, 1996.

[21] Deniau (L.). – Sl++: The scientific library project home page. – 1998. http://home.cern.ch/ ldeniau/html/sl++.html.

[22] Dowd (K.) et Severance (C.R.). – *High Performance Computing.* – O'Reilly & Associated Inc., 1998, second édition, 81–99p.

[23] Eckhaus (W.). – *Studies in Mathematics and its Applications.* – North-Holland Publ. Co., 1979, *Asymptotic Analysis of Singular Perturbations*, volume 9.

[24] Erdelyi (A.). – *Asymptotic expansions*, p. 108. – Dover Publications, New York,Inc. VI, 1956.

[25] Fletcher (C.A.J.). – *Computational Techniques for Fluid Dynamics.* – Berlin: Springer-Verlag, 1998, *Computational Physics*, volume 2.

[26] Funaro (D.), Quarteroni (A.) et Zanolli (P.). – An iterative procedure with interface relaxation for domain decomposition methods. *SIAM J. Num Anal.*, vol. 25, n6, 1988.

[27] Garbey (M.). – Domain decomposition to solve transition layers and asymptotics. *SIAM J. Sci. Comp.*, vol. 15, July 1994, pp. 866–891.

[28] Garbey (M.). – A Schwarz alternating procedure for singular perturbation problems. *SIAM J. Scienc. Comp.*, vol. 17, n5, 1996, pp. 1175–1201.

[29] Garbey (M) et Kaper (H.G.). – Heterogeneous domain decomposition for singularly perturbed elliptic value problems. *SIAM J. Num. Anal.34*, 1997, pp. 1513–1544.

[30] Garbey (M.) et Kuznetsov (Y.). – Parallel Schwarz algorithm for equation with singular perturbed convection diffusion operator. *Preprint UMR5585*, 1996.

[31] Garbey (M.), Viry (L.) et Coulaud (O.). – *Analysis of the Funaro-Quarteroni Procedure for Singularly Perturbed Problems.* – Rapport technique, Elie-Cartan Institute, 1996.

[32] Goplen, B. Ludeking (L.), Smithe (D.) et Warren (G.). – *MAGIC Users Manual.* – Rapport technique n MRC/WDC-R-282, Newington,VA, Mission Research Corp. report, 1991.

[33] Guarba (A.), Mofid (A.) et Peyret (R.). – Spectral solution of free surface flows. *Finite Elements in Analysis and Design*, no331-346, 1994.

[34] Henrot (A.) et Pierre (M.). – About existence of equilibria in electromagnetic casting. *Appl. Math.49*, vol. 3, 1991, pp. 563–575.

[35] Henrot, A. et Pierre (M.). – Un problème inverse en formage des métaux liquides. *Modelisation Math. Anal. Numer. 23*, vol. 1, 1989, pp. 155–177.

[36] Hewett (D.W.), Larson (D.J.) et Doss (S.). – Solution of simultaneous partial differential equations using dynamic adi: solution of the streamlined darwin field equations. *Journal of Computational Physics*, vol. 101, 1992, pp. 11–24.

[37] Standard fortran95, 1997.

[38] Object oriented methods for inter-operable scientific and engineering computing, October 1998.

[39] KAI compilateur C++, Kuck & Associates Inc., 1906 Fox Drive, Champaign,il 61821.

[40] Karmesin (S.), Crotinger (J.), Cummings (J.), Haney (S.), Humphrey (W.), Reynders (J.), Smith (S.) et Williams (T.). – Array design and expression evaluation in POOMA II. – Springer-Verlag, 1998.

[41] Komornik (V.) et Zuazua (E.). – A direct method for the boundary stabilization of the wave equation. *Journal. math. pures et appl.*, vol. 69, 1990, pp. 33–54.

[42] Landau (L.D.) et Lifshitz (E.M.). – *Fluid Mechanics.* – Pergamon press, 1959, *Course of Théorical Physics*, volume 6.

[43] Langdon (A.B.). – On enforcing Gauss'law in electromagnetic particle-in-cell codes. *Computers Physics Communications*, vol. 70, 1992, pp. 447–450.

[44] Li (B.). – The magnetothermal phenomena in electromagnetic levitation processes. *Int Journal Ebgng of Sciences.31*, vol. 2, 1993, pp. 201–220.

[45] Lions (J.L.). – *Quelques méthodes de résolution des problèmes aux limites non linéaires.* – Dunod Gauthier-Villars, Paris, 1969.

[46] Lumsdaine (A.) et Seak (J.). – The Matrix Template Library home page. – 1998. http://www.lsc.nd.edu/research/mtl/index.htm.

[47] Marder (B.). – A method for incorporating Gauss's law into electromagnetic Pic codes. *Journal of Computational Physics*, vol. 68, 1987, pp. 48–55.

[48] Mardhal (P.J.) et Verboncoeur (J.P.). – Charge conservation in electromagnetic PIC codes; spectral comparison of Boris/Dadi and Langdon-Marder methods. *Computers Physics Communications*, vol. 106, 1997, pp. 219–229.

[49] Mestel (A.J.). – Magnetic levitation of liquids metals. *Journal of Fluid Mechanics 117*, 1982, pp. 27–43.

[50] Morse (R.L.). – *"Multidimensional Plasma Simulation by the PIC Method" in Methods of Computational Physics*, p. 213. – Academic Press, New York, 1970.

[51] Morse (R.L.) et Nielson (C.W.). – *Physics Fluids*, vol. 14, 1971, p. 830.

[52] Munz (C-D.), Schneider (R.), Sonnendrucker (E.) et Voss (U.). – Maxwell's equations when the charge conservation is not satisfied. *C.R. Acad. Sci Paris*, vol. 328, 1999, pp. 431–436.

[53] Munz (C.D.), Omnes (P.), Schneider (R.), Sonnendrucker (E.) et Voss (U.). – Divergence correction techniques for Maxwell solvers based on a hyperbolic model. *Journal of Computers in Physics*, 1999.

[54] Musser (D.). – A Standard Template Library Home Page. – 1994. http://www.cs.rpi.edu/ musser/stl.html.

[55] Musser (David R.) et Stepanov (Alexander A.). – Algorithm-oriented generic libraries. *Software: Practice and Experience*, vol. 24, n7, July 1994, pp. 632–642.

[56] Nielsen (D.E.) et Drobot (A.T.). – An analysis and optimization of the pseudo-current method. *Journal of Computational Physics*, vol. 89, 1990, pp. 31–40.

[57] Novelect. – *Les applications innovantes de l'induction dans l'industrie. Les guides de l'innovation.* – Rapport technique, EDF, 1992.

[58] O'Malley (R.E.). – *Singular perturbation methods for ordinary differential equations.* – Applied Mathematical Sciences. 89. New York etc.: Springer-Verlag,, 1991.

[59] Pierre (M.) et Brancher (J-P.). – Control of free surfaces of liquid metals by a magnetic field: Modelling, analysis and applications. *Eur. J. Mech, B 10*, vol. 5, 1991, pp. 443–448.

[60] Pierre (M.) et Roche (J-R.). – Computation of free surfaces in the electromagnetic shaping of liquid metals by optimization algorithms. *Eur. J. Mech, B 10*, vol. 5, 1991, pp. 489–500.

[61] Robison (Arch D.). – The abstraction penalty for small objects in C++. *In: POOMA'96: The Parallel Object-Oriented Methods and Applications Conference.* – Feb. 28 - Mar. 1 1996. Santa Fe, New Mexico.

[62] Robison (Arch D.). – C++ gets faster for scientific computing. vol. 10, 1997, pp. 458–462.

[63] Roche (J. R.) et Pierre (M.). – Numerical simulation of tridimensional electromagnetic shaping of liquid metals. *Numer. Math. 65*, vol. 2, 1993, pp. 203–217.

[64] Shercliff (J.A.). – A magnetic shaping of molten metal columns. *Proc. Royal. Soc. London 375,A*, 1981, pp. 455–473.

[65] Siek (J.G.). – *A modern Framework for Portable High Performance Numerical Linear Algebra.* – Thèse de PhD, University of Notre Dame, March 1999.

[66] Sneyd (A.D.). – Fluid flow induced by a rapidly alternating or rotating magnetic field. *Journal of Fluid Mechanics 92*, 1979, pp. 35–51.

[67] Sneyd (A.D.) et Mofatt (H.K.). – Fluid dynamical aspects of the levitation-melting process. *Journal of Fluid Mechanics 117*, 1982, pp. 45–70.

[68] Stepanov (Alex). – Stepanov's abstraction penalty benchmark,?? Need a citation for this.

[69] Stroustrup (Bjarne). – *The C++ programming language.* – Addison-Wesley, 1991, 2nd édition.

[70] Temam (R.). – *Navier-Stokes equation.* – North-Holland, 1977.

[71] Unruh (Erwin). – Prime number computation, 1994. ANSI X3J16-94-0075/ISO WG21-462.

[72] Veldhuizen (T.). – Expression templates. *C++ Report Magazine*, vol. 7, n5, 1995, pp. 26–31.

[73] Veldhuizen (Todd). – Using C++ template metaprograms. *C++ Report*, vol. 7, n4, mai 1995, pp. 36–43. – Reprinted in C++ Gems, ed. Stanley Lippman.

[74] Veldhuizen (Todd L.). – Scientific computing: C++ versus Fortran: C++ has more than caught up. *Dr. Dobb's Journal of Software Tools*, vol. 22, n11, novembre 1997, pp. 34, 36–38, 91.

[75] Veldhuizen (Todd L.). – Arrays in blitz++. *In : Proceedings of the 2nd International Scientific Computing in Object-Oriented Parallel Environments (ISCOPE'98).* – Springer-Verlag, 1998.

[76] Veldhuizen (Todd L.). – *The Blitz++ User Guide.* – 1998. http://-seurat.uwaterloo.ca/blitz/.

[77] Veldhuizen (Todd L.). – C++ templates as partial evaluation. *In : Proceedings of the ACM SIGPLAN Workshop on Partial Evaluation and Semantics-Based Program Manipulation.* pp. 13–18. – BRICS, 1999.

[78] Veldhuizen (Todd L.) et Gannon (Dennis). – Active libraries: Rethinking the roles of compilers and libraries. *In : Proceedings of the SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing (OO'98).* – SIAM Press, 1998.

[79] Villasenor (J.) et Buneman (O.). – Rigorous charge conservation for local electromagnetic field solvers. *Journal of Computational Physics*, vol. 69, 1992, pp. 306–316.

[80] Weber (J.C.), Buxmann (K.), Von Kaenel (R.) et Plata (M.). – *New Applications for Electromagnetic Casting Process, Lights Metal*, pp. 503–507. – 1988.

Madame **VIRY Laurence**

DOCTORAT de l'UNIVERSITE HENRI POINCARE, NANCY-I

en MATHEMATIQUES APPLIQUEES

VU, APPROUVÉ ET PERMIS D'IMPRIMER

Nancy, le 13 janvier 2000 n° 345

Le Président de l'Université

UNIVERSITÉ HENRI POINCARÉ, NANCY 1 — LE PRÉSIDENT

C. BURLET

3C-F9

UNIVERSITÉ HENRI POINCARÉ - NANCY 1
24-30, rue Lionnois B.P. 3069 54013 Nancy cedex - Tél. 83 85 48 00 - Fax 83 85 48 48

Adresse électronique : ———————— @presn1.u-nancy.fr