# Méthodes de décomposition de domaine pour la formulation mixte duale du problème critique de la diffusion des neutrons

Pierre Guérin

**THÈSE DE DOCTORAT DE L'UNIVERSITÉ PARIS VI**

Spécialité : Mathématiques appliquées

Présentée par

**Pierre GUÉRIN**

Pour obtenir le grade de

**DOCTEUR DE L'UNIVERSITÉ PARIS VI**

Sujet de la thèse

# Méthodes de décomposition de domaine pour la formulation mixte duale du problème critique de la diffusion des neutrons

soutenue le lundi 3 décembre 2007 devant le jury composé de

| | |
|---|---|
| **Yvon Maday** | Directeur de thèse |
| **Damien Tromeur-Dervout** | Rapporteur |
| **Ernest H. Mund** | Rapporteur |
| **Grégoire Allaire** | Président |
| **Frédéric Nataf** | |
| **François-Xavier Roux** | |
| **Jean-Jacques Lautard** | |

**Laboratoire de rattachement :**

Laboratoire de Logiciels pour la Physique des Réacteurs
Service d'Études des Réacteurs et de Mathématiques Appliquées
Département de Modélisation des Systèmes et Structures
Direction des Activités Nucléaires de Saclay
Direction de l'Énergie Nucléaire
Commissariat à l'Énergie Atomique

CEA de Saclay
DEN/DANS/DM2S/SERMA/LLPR
91191 Gif sur Yvette CEDEX

# Domain decomposition methods for the mixed dual formulation of the critical neutron diffusion problem

**Pierre GUÉRIN**

## Abstract

The neutronic simulation of a nuclear reactor core is performed using the neutron transport equation, and leads to an eigenvalue problem in the steady-state case. Among the deterministic resolution methods, diffusion approximation is often used. For this problem, the MINOS solver based on a mixed dual finite element method has shown his efficiency. In order to take advantage of parallel computers, and to reduce the computing time and the local memory requirement, we propose in this dissertation two domain decomposition methods for the resolution of the mixed dual form of the eigenvalue neutron diffusion problem. The first approach is a component mode synthesis method on overlapping subdomains. Several eigenmodes solutions of a local problem solved by MINOS on each subdomain are taken as basis functions used for the resolution of the global problem on the whole domain. The second approach is a modified iterative Schwarz algorithm based on non-overlapping domain decomposition with Robin interface conditions. At each iteration, the problem is solved on each subdomain by MINOS with the interface conditions deduced from the solutions on the adjacent subdomains at the previous iteration. The iterations allow the simultaneous convergence of the domain decomposition and the eigenvalue problem. We demonstrate the accuracy and the efficiency in parallel of these two methods with numerical results for the diffusion model on realistic 2D and 3D cores.

**Key words:** domain decomposition, eigenvalue problem, parallel computation, diffusion equation, neutronics, mixed dual finite elements.

# Méthodes de décomposition de domaine pour la formulation mixte duale du problème critique de la diffusion des neutrons

**Pierre GUÉRIN**

## Résumé

La simulation de la neutronique d'un cœur de réacteur nucléaire est basée sur l'équation du transport des neutrons, et un calcul de criticité conduit à un problème à valeur propre. Parmi les méthodes de résolution déterministes, l'approximation de la diffusion est souvent utilisée. Le solveur MINOS basé sur une méthode d'éléments finis mixte duale, a montré son efficacité dans la résolution de ce problème. Afin d'exploiter les ordinateurs parallèles, et de réduire les coûts en temps de calcul et en mémoire, nous proposons dans ce mémoire deux méthodes de décomposition de domaine pour la résolution du problème à valeur propre de la diffusion des neutrons sous forme mixte duale. La première méthode est inspirée d'une méthode de synthèse modale : la solution est cherchée dans une base constituée d'un nombre fini de modes propres locaux calculés par MINOS sur des sous-domaines recouvrants. La deuxième méthode est un algorithme itératif de Schwarz modifié qui utilise des sous-domaines non recouvrants et des conditions de Robin aux interfaces entre sous-domaines. A chaque itération, le problème est résolu par MINOS sur chaque sous-domaine avec des conditions aux interfaces calculées à partir des solutions sur les sous-domaines adjacents à l'itération précédente. Les itérations permettent la convergence simultanée de la décomposition de domaine et du problème à valeur propre. Les résultats numériques obtenus sur des modèles 2D et 3D de cœurs réalistes montrent la précision et l'efficacité en parallèle de ces deux méthodes.

**Mots clés :** décomposition de domaine, problème à valeur propre, calcul parallèle, équation de la diffusion, neutronique, éléments finis mixtes duaux.

# Remerciements

En premier lieu, je tiens à remercier Jean-Jacques Lautard, qui m'a encadré au CEA Saclay pendant mes trois ans de thèse. Ses conseils avisés, les réponses qu'il a apportées à mes nombreuses questions, son soutien et son enthousiasme constants ont été déterminants dans l'avancée de ma thèse. Travailler avec lui aura été très agréable et très enrichissant.

Je voudrais ensuite adresser toute ma reconnaissance à Yvon Maday, qui a accepté de diriger ma thèse. L'expertise qu'il a apportée sur les problèmes théoriques et numériques que j'ai rencontrés durant mon travail a été essentielle pour le bon déroulement de ma thèse.

J'exprime aussi ma gratitude à Ernest Mund et Damien Tromeur-Dervout qui ont accepté d'être rapporteurs et de relire mon manuscrit avec attention. Je remercie également Grégoire Allaire d'avoir présidé le jury, ainsi que Frédéric Nataf et François-Xavier Roux qui ont bien voulu en être membres.

J'ai beaucoup apprécié pendant mes trois années de thèse l'environnement de travail au CEA, et je tiens à remercier le Service d'Études des Réacteurs et de Mathématiques Appliquées avec lequel je continuerai à avoir des relations de travail ; j'y ai rencontré des collègues qui sont maintenant des amis. Je remercie en particulier Anne-Maire Baudron pour l'aide précieuse qu'elle m'a prodiguée pour la compréhension du solveur MINOS. J'ai aussi une pensée particulière pour Gabriele, François, Steve et Pietro qui ont préparé leur thèse en même temps que moi, et avec qui j'ai eu de nombreuses discussions très enrichissantes.

Je remercie mes collègues d'EDF du département SINETICS : je leur suis reconnaissant de m'avoir aidé à finir ma thèse dans de bonnes conditions tout en commençant mon nouveau travail. Les échanges d'idées que nous avons eus m'ont permis d'améliorer significativement ma soutenance.

Mes remerciements vont enfin à ma famille, en particulier à mes parents, et à mes proches qui m'ont toujours soutenu dans mon travail. Leurs encouragements et leur réconfort dans les moments difficiles m'ont été indispensables pour persévérer et achever mon travail. Je remercie ceux d'entre eux qui ont pu se libérer pour être présents à ma soutenance, et qui ont contribué à rendre ce moment très convivial et unique en réunissant en même temps mes collègues, mes amis et ma famille.

**Merci à vous tous**.

# Table des matières

# Introduction

Les défis en matière de Recherche et Développement pour les centrales nucléaires sont grands. En effet, les industriels souhaitent améliorer le rendement du parc de centrales actuel, et parallèlement de nouveaux concepts de réacteurs doivent être trouvés. La simulation numérique joue un rôle prépondérant pour répondre à ces défis. En particulier, la neutronique, qui consiste en l'étude du cheminement des neutrons, fait intervenir des modèles numériques complexes. Une étape importante de la neutronique est le calcul du flux de neutrons sur le cœur du réacteur. Il est assuré par des solveurs de cœur , basés sur des approximations de l'équation du transport des neutrons, comme le modèle de la diffusion. Ces solveurs doivent être de plus en plus performants. En effet ils sont utilisés par les industriels qui souhaitent obtenir le flux de neutrons rapidement en tenant compte de la complexité croissante des modèles et des géométries.

Le solveur de cœur MINOS actuellement utilisé par le CEA permet entre autre de résoudre la formulation mixte duale du problème à valeur propre de la diffusion. Il offre d'excellentes performances, mais un handicap est qu'il est difficilement parallélisable. Hors la parallélisation des codes de calcul permet d'exploiter les calculateurs modernes composés d'un grand nombre de processeurs, et ainsi de réduire les temps de calcul.

Les méthodes de décomposition de domaine sont souvent envisagées pour paralléliser un solveur. En effet les calculs locaux sur chaque sous-domaine sont effectués par un solveur séquentiel, et peuvent être répartis sur plusieurs processeurs. Les couplages entre les sous-domaines sont alors assurés par des communications entre les processeurs.

Plusieurs méthodes de décomposition de domaine ont été proposées pour le solveur MINOS. Dans la thèse [Coulomb, 1989], une méthode de Jacobi par blocs est développée à la fois pour la formulation primale et pour la formulation mixte duale du problème de la diffusion. Mais son efficacité en parallèle n'a pas été testée et une interface avec MINOS semble délicate à mettre en œuvre . Une autre méthode, développée dans la première partie de la thèse [Pinchedez, 1999], consiste à répartir les données sur les différents processeurs, et à effectuer les calculs localement sur ces données. Cette approche offre l'avantage d'être facilement intégrable au solveur MINOS. Mais là aussi les performances en parallèle se dégradent à mesure que le nombre de processeurs augmente, car les communications entre processeurs deviennent trop importantes. Dans une deuxième partie de cette thèse, une méthode de synthèse modale pour le problème à valeur propre de la diffusion sous forme primale est présentée. Plusieurs modes propres sont calculés sur des sous-domaines non recouvrants, et sont utilisés comme fonctions de base pour la résolution du problème sur le domaine complet. L'avantage de cette méthode est qu'elle permet à priori d'obtenir une parallélisation efficace. Mais l'utilisation de sous-domaines sans recouvrement rend la mise en œuvre de la méthode complexe, car elle nécessite l'introduction de modes d'interface. Seuls des résultats sur des géométries homogènes ont été obtenus, et la méthode n'est pas adaptée à la formulation mixte duale utilisée par MINOS.

Ces études apportent donc des solutions intéressantes pour paralléliser la résolution du problème de la diffusion des neutrons. Mais l'une d'entre elles est pénalisée par une chute de l'efficacité quand le nombre de processeurs augmente, et les deux autres sont difficilement intégrables au solveur MINOS.

**Notre étude consiste à développer une méthode de décomposition de domaine pour la résolution du problème à valeur propre de la diffusion des neutrons sous forme mixte duale.** Elle doit être intégrée au projet APOLLO3/DESCARTES en cours de développement au CEA, en utilisant le solveur MINOS pour les calculs locaux sur les sous-domaines.

Nous commençons ce manuscrit par une présentation du contexte de notre étude. Le chapitre 1 est consacré au fonctionnement d'un réacteur nucléaire, à la neutronique et aux méthodes numériques utilisées dans cette discipline. Le chapitre 2 décrit en détail le solveur MINOS et son intégration dans le projet APOLLO3/DESCARTES. Des rappels sur la parallélisation sont faits au chapitre 3, et deux méthodes de décomposition de domaine sont présentées : la méthode de synthèse modale et l'algorithme de Schwarz. Le chapitre se termine par une présentation des méthodes de parallélisation de MINOS développées dans [Coulomb, 1989] et [Pinchedez, 1999].

Les chapitres suivants sont dédiés à l'exposé de la démarche de notre étude. Dans un premier temps, notre avons poursuivi et étendu le travail de Katia Pinchedez sur la synthèse modale pour le problème à valeur propre de la diffusion. Contrairement à l'approche qu'elle propose, nous avons opté pour des sous-domaines qui se recouvrent. Par ailleurs, notre objectif a été d'adapter la méthode à un problème mixte, pour l'appliquer à la formulation mixte duale du problème à valeur propre de la diffusion. Puis notre but a été d'analyser la qualité d'approximation en fonction de la taille du recouvrement, de la configuration des sous-domaines et du nombre de modes propres locaux utilisés sur chaque sous-domaine. Pour cela, des tests ont été réalisés sur un modèle simplifié en 1D (Chapitre 4). Afin de valider la méthode sur des cas réalistes en 2D et 3D, un programme a été de développé en utilisant le solveur MINOS pour les calculs locaux. La parallélisation de l'algorithme a ensuite été étudiée (Chapitre 5). Pour réduire le coût des calculs locaux sur les sous-domaines, nous nous sommes appuyés sur un théorème d'homogénéisation afin de calculer uniquement le premier mode propre sur chaque sous-domaine, et de remplacer les modes d'ordre supérieurs par des fonctions factorisées. Des tests sur le modèle 1D ont été menés pour comparer cette méthode de synthèse modale factorisée à la méthode précédente. Des calculs sur des géométries complexes ont été faits séquentiellement afin de vérifier si la qualité d'approximation est satisfaisante, puis en parallèle pour mesurer l'efficacité de la méthode et les temps de calcul (Chapitre 6). La méthode de synthèse modale et sa variante factorisée ont fait l'objet de l'article [Guerin *et al.*, 2007a]. Dans un deuxième temps les limitations de la méthode de synthèse modale nous ont amené à développer une méthode de décomposition de domaine itérative. Notre objectif a été d'adapter un algorithme itératif de type Schwarz à la résolution du problème à valeur propre de la diffusion sous forme mixte duale. Nous avons fait le choix d'une décomposition de domaine sans recouvrement, et de conditions de Robin aux interfaces des sous-domaines. Afin d'évaluer l'impact sur la convergence du nombre de sous-domaines et de la valeur des coefficients de Robin aux interfaces, des tests en 1D ont été réalisés (Chapitre 7). Nous avons alors voulu vérifier l'efficacité de la méthode en parallèle sur des géométries réalistes. Pour cela, l'algorithme a été implémenté dans APOLLO3/DESCARTES en apportant quelques modifications au solveur MINOS pour assurer les communications aux interfaces (Chapitre 8). Cette méthode est présentée dans un article soumis pour la revue MATCOM.

Première partie

# Présentation du contexte physique et des méthodes mathématiques

# Chapitre 1

# Présentation du fonctionnement d'un réacteur nucléaire et de la neutronique

---

*Ce chapitre a pour but de donner quelques bases de la physique d'un réacteur nucléaire. Nous aborderons d'abord le fonctionnement général d'un réacteur, en donnant des précisions dans le cas particulier des réacteurs à eau sous pression. Nous introduirons ensuite la neutronique d'un cœur de réacteur, c'est à dire l'étude de la population de neutrons. Enfin nous présenterons les méthodes numériques pour les calculs de neutronique, qui permettent d'élaborer des schémas de calcul à la fois précis et efficaces. Ce chapitre s'inspire notamment de [Reuss, 2003] et de [Bussac et Reuss, 1985], qu'on pourra consulter pour avoir plus de détails.*

---

## 1.1 Fonctionnement d'un réacteur nucléaire

Le principe d'une centrale nucléaire est commun à la plupart des centrales électriques : une source de chaleur permet de vaporiser de l'eau, et la vapeur est utilisée pour alimenter des turbines reliées à un alternateur producteur d'électricité. La particularité d'un réacteur nucléaire est que son fonctionnement est basé sur la notion de réaction en chaîne de fission atomique. Des neutrons libres entrent en collision avec des atomes du matériau combustible. Le choc entraîne la fission du noyau de certains de ces atomes, libérant de nouveaux électrons libres qui engendrent à leur tour des fissions (figure 1.1). La réaction de fission est exothermique, et la chaleur libérée est utilisée pour produire de l'électricité.

Le comportement global de la réaction en chaîne est lié au facteur de multiplication effectif $k_{eff}$. En effet il détermine son évolution : $N$ fissions engendrent $Nk_{eff}$ fissions, produisant elles-mêmes $Nk_{eff}^2$ fissions. Si $k_{eff} < 1$, la réaction est dite sous-critique, elle tend à s'éteindre. Si $k_{eff} > 1$, la réaction est dite sur-critique et elle tend à s'emballer. Une réaction stable dite critique est caractérisée par un $k_{eff} = 1$ : dans ce cas elle s'auto-entretient à un niveau constant.

La maîtrise de la réactivité est essentielle, car elle permet de contrôler le réacteur, de modifier sa puissance, d'en assurer la stabilité et la sûreté. Le contrôle de la réaction s'effectue en modifiant momentanément la valeur du $k_{eff}$ autour de 1. Le démarrage ou l'augmentation de puissance du réacteur est obtenu en le rendant temporairement sur-critique, tandis qu'il doit être sous-critique pour son arrêt ou une diminution de puissance.

Le contrôle de la réactivité se fait notamment par l'intermédiaire de grappes de commande dans le cœur . Ces grappes, aussi appelées barres de contrôle, sont constituées de matériaux dits absorbants qui capturent les neutrons. Leur introduction dans le cœur permet de faire chuter la réactivité, alors que leur retrait permet de l'augmenter. L'utilisation de poisons consommables tel que le bore permet aussi un ralentissement de la réaction en chaîne. Par ailleurs, une auto-régulation du système est induite par des contre-réactions liées aux variations de température, car elles ont tendance à faire baisser le facteur de multiplication si la puissance augmente.

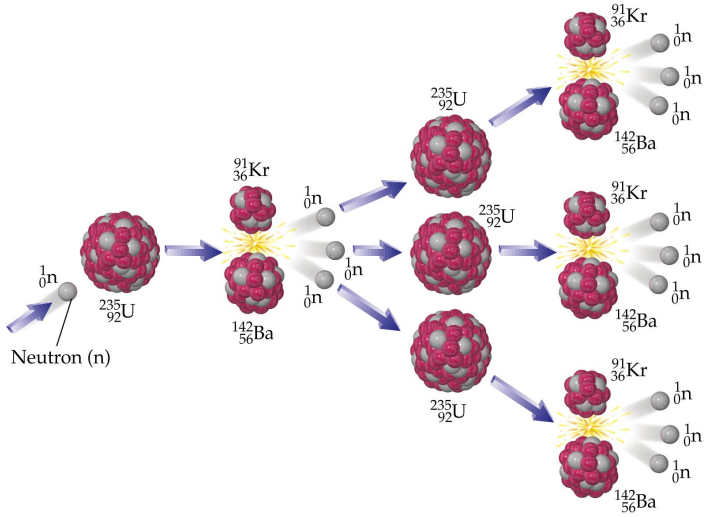



FIG. 1.1: Réaction en chaîne de fission.

Une filière de réacteur est caractérisée par le choix de certains composants :

- le combustible, dont les atomes assurent la réaction de fission. A l'état naturel seul l'uranium fissionne par choc avec des neutrons libres. Des deux isotopes 238 et 235, seul l'uranium 235 est susceptible de fissionner facilement par choc avec des neutrons incidents. Mais le taux de l'isotope 235 est très faible dans l'uranium naturel, il est donc souvent nécessaire de l'enrichir pour augmenter ce taux.
- Le caloporteur circule dans le cœur du réacteur et absorbe la chaleur dégagée par la réaction en chaîne. Cette chaleur permet ensuite de générer de l'électricité.
- Le modérateur assure le ralentissement des neutrons, afin de les amener à une énergie cinétique faible augmentant la probabilité de fission de l'uranium 235.

Deux concepts de réacteur se distinguent :

- les réacteurs à neutrons thermiques.
  Les neutrons issus de la fission sont dits "rapides", car leur énergie cinétique est grande. La présence d'un modérateur dans le cœur permet de les ralentir et de les "thermaliser", augmentant ainsi la probabilité qu'ils engendrent une fission de l'uranium 235. Ce type de réacteur peut donc se contenter d'un combustible peu enrichi, pauvre en isotopes fissiles. La quasi-totalité du parc nucléaire français est composé de réacteurs à neutrons thermiques : les réacteurs à eau sous pression (REP, PWR en anglais pour *Pressurized Water Reactor*), décrits ci-dessous.
- Les réacteurs à neutrons rapides.
  Ces réacteurs ne sont pas modérés : ils utilisent pour la fission les neutrons à l'énergie à laquelle ils sont produits. Comme la probabilité que ces neutrons rapides engendrent des fissions est faible, le combustible doit être fortement enrichi en isotopes fissiles comme l'uranium 235 ou le plutonium 239. Seul un réacteur de ce type est actuellement en fonctionnement en France : le réacteur PHENIX, SUPER PHENIX étant démantelé. Mais les réacteurs à neutrons rapides sont toujours à l'étude, notamment dans le cadre des projets de réacteurs de Génération IV.

### 1.1.1 Le réacteur à eau sous pression (REP)

La filière des réacteurs à eau sous pression mérite quelques explications supplémentaires, car elle a été choisie pour constituer le parc nucléaire français actuel. La maintenance et l'exploitation de ces réacteurs est un enjeu crucial pour les acteurs de l'énergie nucléaire en France (notamment le CEA, EDF et AREVA). L'un des objectifs est d'augmenter la durée de vie de ces réacteurs dans des conditions de sécurité et de rendement optimisées. Les besoins en simulation numérique pour ces réacteurs sont grands, à la fois pour leur exploitation (calcul des nappes de puissance en temps réel, chargement du combustible...) et pour leur optimisation (études des combustibles, augmentation du rendement, études de sûreté et de situations accidentelles...).

La principale caractéristique des REP est d'utiliser l'eau du circuit primaire à la fois comme modérateur et comme caloporteur. Le combustible utilisé dans les REP est de l'oxyde d'uranium faiblement enrichi. Le combustible MOX, composé d'oxyde d'uranium et de plutonium est également utilisé dans certaines centrales, et permet de recycler le plutonium issu du retraitement du combustible irradié.

La figure 1.2 présente le fonctionnement du réacteur. Le cœur est enfermé dans une cuve, et la chaleur est évacuée par le circuit primaire d'eau maintenue liquide par une forte pression à l'aide d'un pressuriseur (150 bars, 320 °C en sortie de cœur ). L'eau quittant le cœur est répartie entre trois ou quatre boucles comportant chacune une pompe primaire de recirculation

et un générateur de vapeur. Ce dernier permet de vaporiser l'eau du circuit secondaire, et la vapeur alimente des turbines reliées à un alternateur producteur d'électricité. La vapeur passe ensuite dans un condenseur, et l'eau en sortie est réinjectée par des pompes dans les générateurs de vapeur. Le condenseur est refroidi par un troisième circuit d'eau, prélevé dans un fleuve ou dans la mer, et passant éventuellement dans un aéroréfrigérant. Le rendement global d'un REP est assez faible, de l'ordre de 33% : 3 joules libérés par fission engendrent 1 joule électrique et 2 joules de chaleur dispersés dans l'environnement. La géométrie du

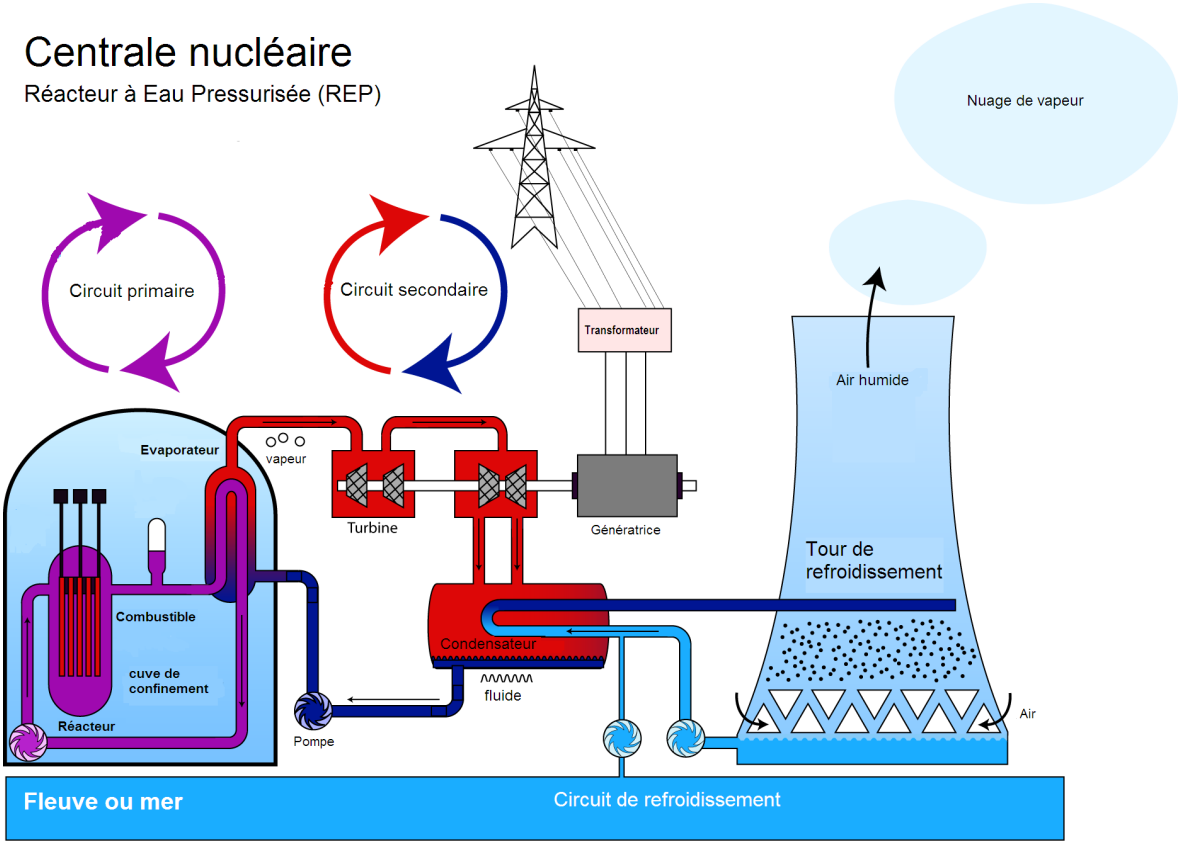



FIG. 1.2: Schéma du fonctionnement d'un réacteur à eau sous pression.

cœur d'un REP est multi-échelles (figure 1.3). Elle est constituée d'une juxtaposition de 157 assemblages de combustible (figure 1.4), disposés verticalement de façon à former un ensemble approximativement cylindrique. Chaque assemblage combustible est organisé en un réseau orthogonal de 17 par 17 crayons. Les 264 crayons combustibles sont constitués d'un tube métallique contenant un empilement de pastilles d'oxyde d'uranium. L'emplacement central est réservé à un tube servant à l'instrumentation du cœur ; les 24 positions restantes sont occupées par des tubes guides dans lesquelles viennent s'insérer les grappes annexes : grappes de contrôle, grappes de poison consommable, grappes source de neutrons. L'ensemble carré formé d'un crayon et du modérateur qui l'entoure est appelé cellule, et est souvent utilisé dans les calculs de cœur comme maille géométrique, avec des sections efficaces constantes homogénéisées.

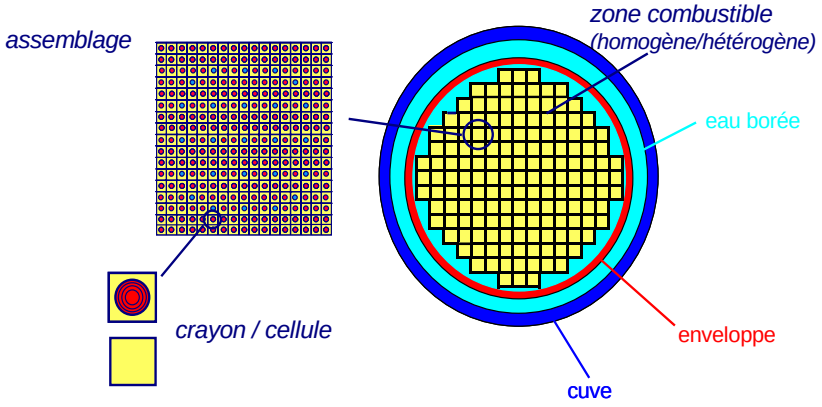
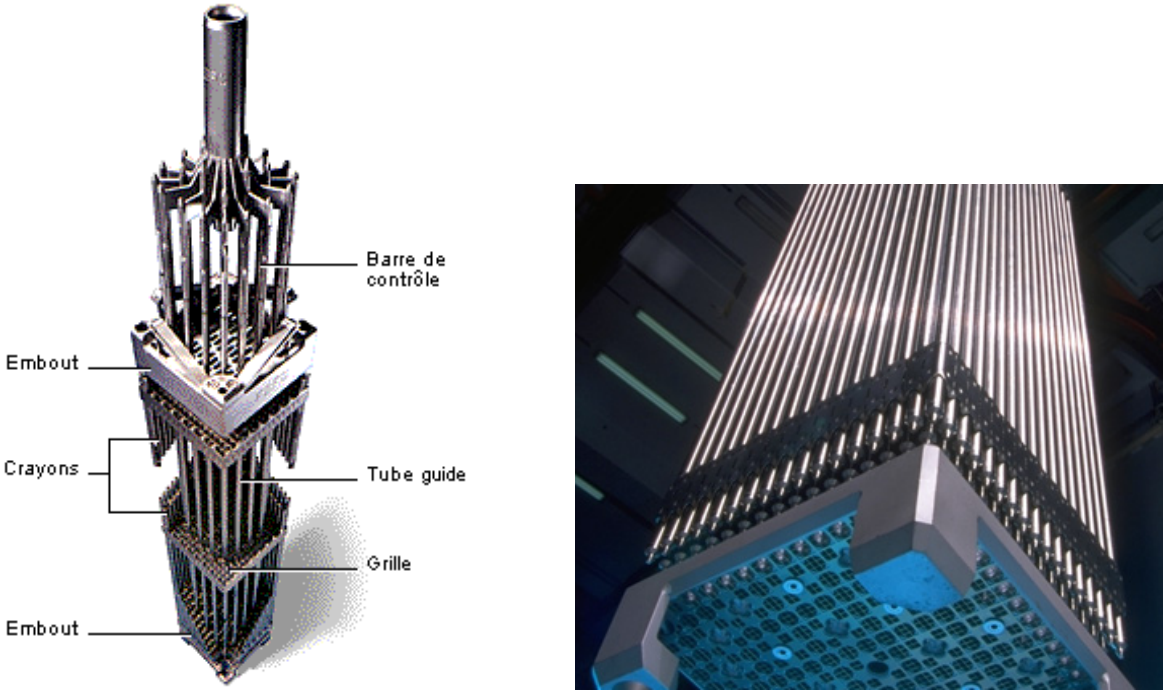



FIG. 1.3: Géométrie multi-échelles d'un réacteur à eau sous pression.




(a) Constitution

(b) Assemblage MOX

FIG. 1.4: Assemblage combustible d'un réacteur à eau sous pression.

## 1.2 La neutronique

"La neutronique est l'étude du cheminement des neutrons dans la matière et des réactions qu'ils y induisent, en particulier la génération de puissance par la fission de noyaux d'atomes lourds." ([Reuss, 1998])

La population des neutrons peut être représentée par le flux neutronique $\Phi = nv$. $n$ est la densité des neutrons, c'est à dire le nombre de neutrons par unité de volume, et $v$ est leur vitesse. Ce flux caractérise les neutrons "qui voyagent", c'est à dire pris en vol depuis leur point de départ jusqu'à leur prochaine collision. Remarquons que le flux neutronique n'est pas un flux au sens usuel, qui décrit une quantité traversant une surface.

Si l'on veut représenter complètement (de façon statistique) une population de neutrons dans un système, sept variables sont nécessaires :

- trois variables d'espace (x, y, z) pour repérer la position $\overrightarrow{r}$ d'un neutron ;
- trois variables de vitesse $(v, \theta, \varphi)$, notées $(v, \overrightarrow{\Omega})$, pour repérer leur état ;
- une variable de temps (t) pour préciser l'instant de l'observation.

Le flux neutronique est donc une fonction qui dépend de ces sept variables.

Une autre donnée fondamentale des phénomènes de neutronique est l'ensemble des probabilités d'interaction des neutrons avec les différents noyaux cibles. Les sections efficaces sont les grandeurs caractéristiques de ces probabilités. Une section efficace peut-être microscopique, c'est à dire caractéristique d'une cible individuelle, ou macroscopique, c'est à dire caractéristique d'un matériau contenant un grand nombre de cibles. Pour un noyau donné, les sections microscopique $\sigma$ et macroscopique $\Sigma$ sont liées par la relation : $\Sigma = N\sigma$, où N est égal au nombre de noyaux par unité de volume.

### 1.2.1 L'équation de Boltzmann

Le bilan neutronique est quant à lui décrit par l'équation de Boltzmann pour le flux neutronique. Deux opérateurs figurent dans cette équation : l'opérateur de transport, qui modélise les neutrons cheminant en ligne droite sans interaction avec la matière, et l'opérateur de collision, qui correspond aux interactions avec les particules.

L'opérateur de transport peut-être écrit soit sous forme intégrale, soit sous forme intégro-différentielle (intégrale pour v et $\overrightarrow{\Omega}$, différentielle pour $\overrightarrow{r}$ et t). Bien que les deux formulations soient mathématiquement équivalentes, elles conduisent à des méthodes de résolution numérique différentes. Nous présentons ici la forme intégro-différentielle, utilisée pour plusieurs types de méthodes déterministes :

$$-\overrightarrow{\nabla}.\left[\overrightarrow{J}(\overrightarrow{r},v,\overrightarrow{\Omega},t)\right]-\Sigma_t(\overrightarrow{r},v,\overrightarrow{\Omega},t)\Phi(\overrightarrow{r},v,\overrightarrow{\Omega},t)+Q(\overrightarrow{r},v,\overrightarrow{\Omega},t)=\frac{1}{v}\frac{\partial}{\partial t}\Phi(\overrightarrow{r},v,\overrightarrow{\Omega},t). \tag{1.1}$$

$\overrightarrow{\nabla}.$ est l'opérateur de divergence, défini par : $\overrightarrow{\nabla}.\overrightarrow{q}=\sum_{i=1}^{d}\frac{\partial q_i}{\partial x_i}$, avec $\overrightarrow{q}=(q_i)_{1\leq i\leq d}$, $d$ étant le nombre de dimensions d'espace. $\overrightarrow{J}$ est le courant, défini par :

$$\overrightarrow{J}(\overrightarrow{r},v,\overrightarrow{\Omega},t)=\overrightarrow{\Omega}\Phi(\overrightarrow{r},v,\overrightarrow{\Omega},t). \tag{1.2}$$

$\Sigma_t$ est la section efficace totale qui correspond aux neutrons pouvant disparaître par choc, c'est à dire par absorption ou par transfert vers une autre vitesse et une autre direction. Le premier terme de l'équation (1.1) correspond à l'opérateur de transport. $Q$ correspond à l'opérateur de collision et se décompose en deux termes que nous allons détailler : $Q = D + S$. D est

l'opérateur de diffusion, qui correspond à la section de diffusion $\Sigma_s$ (ou de transfert, *scattering* en anglais) :

$$D(\overrightarrow{r}, v, \overrightarrow{\Omega}, t) = \int_0^\infty dv' \int_{(4\pi)} d^2\Omega' \Sigma_s \left[\overrightarrow{r}, (v', \overrightarrow{\Omega'}) \rightarrow (v, \overrightarrow{\Omega}), t\right] \Phi(\overrightarrow{r}, v', \overrightarrow{\Omega'}, t). \tag{1.3}$$

$S = S_0 + S_f$ est la source de neutrons. $S_0$ correspond aux sources externes de neutrons, tandis que $S_f$ est la source issue de la fission :

$$S_f(\overrightarrow{r}, v, \overrightarrow{\Omega}, t) = \frac{1}{4\pi}\chi(\overrightarrow{r}, v, \overrightarrow{\Omega}, t) \int_0^\infty dv' \int_{(4\pi)} d^2\Omega' \nu(\overrightarrow{r}, v', \overrightarrow{\Omega'}) \Sigma_f(\overrightarrow{r}, v', \overrightarrow{\Omega'}, t) \Phi(\overrightarrow{r}, v', \overrightarrow{\Omega'}, t). \tag{1.4}$$

$\chi$ est le spectre des neutrons émis par fission, tandis que $\nu$ est le nombre moyen de neutrons émis lors d'une fission. $\Sigma_f$ est la section efficace de fission.

### 1.2.2 Calcul critique

Nous n'évoquerons pas dans notre travail les problèmes de cinétique : nous nous intéresserons uniquement à des régimes stationnaires, qui correspondent à une annulation de la dérivée en temps du flux dans l'équation de Boltzmann (1.1). De plus, nous nous placerons dans des cas où il n'y a pas de sources externes, mais seulement des sources de fission. On se retrouve alors avec un terme source qui dépend du flux, comme l'indique l'équation (1.4). Dans ce cas l'équation de Boltzmann n'a une solution que dans le cas particulier d'un réacteur stationnaire critique : son coefficient de multiplication $k_{eff}$ est égal à 1. Pour contourner cette difficulté et évaluer l'état de stabilité de la réaction en chaîne, on introduit un "paramètre critique" qui modifie fictivement les données du calcul de façon à le rendre critique. Ce paramètre critique n'est autre que le facteur de multiplication effectif $k_{eff}$ introduit dans la section 1.1.

Le calcul critique est alors un problème à valeur propre, dont on cherche le mode fondamental correspondant au flux associé à la plus grande valeur propre $\lambda = k_{eff}$. Ce problème s'écrit (sans les conditions aux bords) :

$$\overrightarrow{\nabla}.\left[\overrightarrow{J}(\overrightarrow{r}, v, \overrightarrow{\Omega})\right] + \Sigma_t(\overrightarrow{r}, v, \overrightarrow{\Omega})\Phi(\overrightarrow{r}, v, \overrightarrow{\Omega}) = D(\overrightarrow{r}, v, \overrightarrow{\Omega}) + \frac{1}{\lambda} S_f(\overrightarrow{r}, v, \overrightarrow{\Omega}). \tag{1.5}$$

### 1.2.3 Les méthodes de résolution numérique

Deux approches sont utilisées pour résoudre numériquement l'équation de Boltzmann (1.1) : l'approche déterministe et l'approche probabiliste.

L'approche probabiliste est de type "Monte-Carlo", et consiste à simuler le plus fidèlement possible le cheminement des neutrons, et après un grand nombre de simulations, à faire un examen statistique des résultats. L'avantage de cette méthode est de résoudre l'équation de Boltzmann sans avoir besoin de la formuler explicitement. Elle permet de traiter le problème en faisant le minimum d'approximation. Mais l'utilisation de cette méthode de résolution coûte très cher en temps de calcul. En effet, l'incertitude statistique d'un résultat diminue comme l'inverse de la racine carrée du nombre de simulations utilisées. Un calcul précis requiert donc un grand nombre de simulations, et donc un temps de calcul élevé. Les codes de calculs Monte-Carlo (comme TRIPOLI4, [Both *et al.*, 2003]) sont pour cela souvent utilisés pour des calculs de référence.

L'approche déterministe est souvent utilisée car elle conduit à des méthodes en général moins coûteuses en temps de calcul que les méthodes probabilistes. On distingue deux types de méthodes déterministes, selon qu'on utilise la formulation intégrale du transport, ou sa

formulation intégro-différentielle. Dans le premier cas la méthode des probabilités de première collision (dite $P_{ij}$) est utilisée, mais nous ne la détaillerons pas.

Quant aux méthodes numériques pour la formulation intégro-différentielle, elles adoptent des approximations sur la variable angulaire et sur la variable d'énergie. On se ramène ainsi à un système d'équations différentielles en espace (si on ne considère pas la variable de temps), puisque les termes intégraux ont été discrétisés.

Deux techniques sont utilisées pour discrétiser la variable angulaire $\overrightarrow{\Omega}$. La première est la technique des ordonnées discrètes, qui discrétise la variable $\overrightarrow{\Omega}$ en N directions. Elle est notamment utilisée par les méthodes $S_N$. Plusieurs méthodes utilisent une deuxième technique qui consiste à faire un développement à un ordre fini de $\overrightarrow{\Omega}$ sur des bases complètes de fonctions :

- les approximations $B_N$ où le terme de transfert $\Sigma_s$ est développé sur la base des harmoniques sphériques jusqu'à l'ordre N ;
- les approximations $P_N$ où le flux est cette fois-ci développé sur la base des harmoniques sphériques ;
- l'approximation $SP_N$, c'est à dire $P_N$ simplifié ([Pomraning, 1993]). Elle présente un intérêt car elle correspond en fait à plusieurs équations de type diffusion couplées. Les solveurs de cœur tel que le solveur MINOS [Baudron et Lautard, 2007] peuvent donc assez facilement résoudre les équations $SP_N$ en gardant le formalisme plus simple de la diffusion. Un autre avantage de l'approximation $SP_N$ est qu'elle offre une meilleure approximation du transport que la diffusion, mais avec des temps de calcul beaucoup moins grands que ceux nécessaires à l'approximation $P_N$. Son défaut est qu'elle ne converge pas vers le transport lorsque $N \to +\infty$, contrairement au $P_N$ ;
- l'approximation de diffusion, détaillée à la section suivante. Elle correspond à l'approximation $P_1$ ou $SP_1$.

La variable d'énergie $E$ (liée à la variable de vitesse $v$) est par contre toujours discrétisée de la même manière, selon la théorie multigroupe. Si $E_0$ est l'énergie maximale que peuvent avoir les neutrons, la variable d'énergie E appartenant à l'intervalle global d'énergie $[0, E_0]$ est discrétisée en $N_g$ groupes. Le groupe g correspond alors à l'ensemble des neutrons d'énergie incluse dans l'intervalle $[E_g, E_{g-1}]$ (les groupes sont numérotés par énergie décroissante). Dans chacun des groupes, le transport des neutrons est traité comme s'ils étaient monocinétiques, c'est à dire comme s'ils avaient tous la même énergie. Les équations propres à chacun des groupes sont couplées car, aux vraies sources émettant dans le groupe considéré, s'ajoutent les taux de transfert dans ce groupe depuis les autres groupes ; et aux vraies absorptions dans ce groupe s'ajoutent les transferts vers les autres groupes.

La qualité d'approximation de ces méthodes dépend des ordres de discrétisation angulaire et énergétique. Plus ils sont élevés, plus la solution se rapproche de la solution exacte, mais plus les calculs sont coûteux en mémoire et en temps de calcul.

### 1.2.4 L'approximation de la diffusion

La grande complexité de l'équation de Boltzmann, due notamment à la dépendance aux sept variables de temps, d'espace, d'angle et de vitesse, fait que sa résolution numérique n'est pas envisageable sur des grands domaines. L'approximation du transport des particules par un opérateur de diffusion est souvent utilisée, car elle permet d'éviter la prise en compte de la variable angulaire $\overrightarrow{\Omega}$. D'autre part, la variation des sections efficaces étant très lente devant le temps de vie moyen des neutrons, on les supposera indépendantes du temps.

L'approximation de diffusion repose sur la loi de Fick, qui exprime l'idée intuitive que les neutrons diffusent des points où ils sont nombreux vers ceux où ils le sont moins, et ceci

d'autant plus que la différence de niveaux de flux est importante. Cette loi exprime que le courant (équation (1.2)) est proportionnel au gradient du flux :

$$\overrightarrow{J}(\overrightarrow{r},v,t) = -D(\overrightarrow{r},v)\overrightarrow{\nabla}\Phi(\overrightarrow{r},v,t). \tag{1.6}$$

D est le coefficient de diffusion, égal en premier approche à $\frac{1}{3\Sigma_t}$.

Mais cette loi n'est valable que si les variations en espace et en temps sont lentes, ce qui implique les hypothèses suivantes :

- les hétérogénéités géométriques sont faibles ;
- la section efficace d'absorption est faible devant la section efficace de diffusion ;
- on ne se place pas trop près des interfaces entre des milieux différents ;
- on ne se place pas trop près des sources concentrées.

L'équation de la diffusion s'obtient en intégrant l'équation de Boltzmann (1.1) par rapport à la variable angulaire $\overrightarrow{\Omega}$, et en remplaçant le courant à l'aide de la loi de Fick (1.6) :

$$\begin{aligned}&\frac{1}{v}\frac{\partial}{\partial t}\Phi(\overrightarrow{r},v,t) - \overrightarrow{\nabla}.\left[D\overrightarrow{\nabla}\Phi(\overrightarrow{r},v,t)\right] + \Sigma_t(\overrightarrow{r},v)\Phi(\overrightarrow{r},v,t)\\ &-\int_0^\infty \Sigma_s(\overrightarrow{r},v'\to v)\Phi(\overrightarrow{r},v',t)dv' = \chi(v)\int_0^\infty \nu(\overrightarrow{r},v')\Sigma_f(\overrightarrow{r},v')\Phi(\overrightarrow{r},v',t)dv' + S_0(\overrightarrow{r},v,t).\end{aligned} \tag{1.7}$$

Dans le cadre d'un calcul critique, introduit section 1.2.2, l'équation de diffusion devient :

$$\begin{aligned}&-\overrightarrow{\nabla}.\left[D\overrightarrow{\nabla}\Phi(\overrightarrow{r},v)\right] + \Sigma_t(\overrightarrow{r},v)\Phi(\overrightarrow{r},v)\\ &-\int_0^\infty \Sigma_s(\overrightarrow{r},v'\to v)\Phi(\overrightarrow{r},v')dv' = \frac{1}{k_{eff}}\chi(v)\int_0^\infty \nu(\overrightarrow{r},v')\Sigma_f(\overrightarrow{r},v')\Phi(\overrightarrow{r},v')dv'.\end{aligned} \tag{1.8}$$

Le traitement numérique de l'équation de diffusion nécessite une discrétisation de la variable d'énergie. De la même manière que pour l'équation de Boltzmann, le traitement multigroupe de l'équation de diffusion aboutit à un système couplé de $N_g$ équations, les $N_g$ inconnues étant les flux pour chaque groupe. L'équation de la diffusion critique pour un groupe $g$ donné s'écrit :

$$\begin{aligned}&-\overrightarrow{\nabla}.\left[D^g\overrightarrow{\nabla}\Phi^g(\overrightarrow{r})\right] + \Sigma_t^g(\overrightarrow{r})\Phi^g(\overrightarrow{r})\\ &-\sum_{g'=1}^{N}\Sigma_s^{g'\to g}(\overrightarrow{r})\Phi^{g'}(\overrightarrow{r}) = \frac{1}{k_{eff}}\chi^g\sum_{g'=1}^{N}\nu^{g'}(\overrightarrow{r})\Sigma_f^{g'}(\overrightarrow{r})\Phi^{g'}(\overrightarrow{r}).\end{aligned} \tag{1.9}$$

On remarque que ce sont les termes liés au transfert et à la fission qui couplent les $N_g$ équations.

L'équation monocinétique de la diffusion est le cas le plus simple où on considère que tous les neutrons ont la même énergie, et s'écrit dans le cas critique sous la forme :

$$-\overrightarrow{\nabla}.\left[D\overrightarrow{\nabla}\Phi(\overrightarrow{r})\right] + \Sigma_a(\overrightarrow{r})\Phi(\overrightarrow{r}) = \frac{1}{k_{eff}}\nu(\overrightarrow{r})\Sigma_f(\overrightarrow{r})\Phi(\overrightarrow{r}), \tag{1.10}$$

où $\Sigma_a = \Sigma_t - \Sigma_s$ est la section efficace d'absorption. La théorie à un groupe fournit des résultats qualitativement corrects. Le passage à la théorie multigroupe ne pose pas de problème particulier, car les termes qui couplent les équations (1.9) en énergie sont traités numériquement comme des sources.

### 1.2.5 Schéma de calcul

Un calcul complet et précis de la neutronique d'un cœur de réacteur nucléaire est très complexe, car il doit prendre en compte toutes les hétérogénéités au niveau spatiale et dans les données nucléaires (les sections efficaces). La résolution directe de l'équation de Boltzmann sur le cœur complet avec une discrétisation fine est inaccessible en temps de calcul, surtout pour les industriels et les chargés d'étude qui doivent simuler un grand nombre de cas. Des schémas de calcul, c'est à dire des calculs successifs correspondant à des approximations à plusieurs niveaux, sont donc développés pour obtenir des résultats très précis, avec des temps de calcul raisonnables.

Les données nucléaires de base sont issues de nombreuses mesures expérimentales, et permettent de constituer des bibliothèques de sections efficaces qui décrivent très finement les propriétés neutroniques d'un grand nombres de noyaux. Les différentes phases de calcul vont permettre d'homogénéiser en espace et de condenser en énergie ces sections efficaces, en fonction de la configuration du cœur à calculer. Ces sections "grossières" adaptées permettront de réaliser un calcul de cœur basé sur un modèle et des données très simplifiés, tout en apportant une précision satisfaisante.

Un schéma de calcul standard repose principalement sur une chaîne de trois calculs distincts, qui correspondent à trois niveaux d'approximation et de détail dans la physique neutronique :

- la première étape, correspondant à l'échelle la plus fine, permet de traiter des phénomènes complexes, dits d'"autoprotection", dus à la présence de nombreuses résonances des noyaux lourds, c'est à dire des variations très fortes et très nombreuses des sections efficaces en fonction de l'énergie. Les deux principales opérations à réaliser dans cette première phase de calcul consistent à tabuler les grandeurs caractéristiques des phénomènes de résonance, et à réaliser une première condensation énergétique pour les calculs de transport à l'étape suivante (environ une centaine de groupes). Ces calculs sont faits en principe une fois pour toute.
- La phase de calcul suivante consiste à réaliser sur chaque assemblage un calcul de transport à deux dimensions d'espace utilisant les sections autoprotégées. Ces calculs prennent en compte les hétérogénéités spatiales fines : combustible, gaine, modérateur... Cette étape permet de condenser à nouveau les sections efficaces en énergie, et de réaliser une homogénéisation spatiale. Il en résulte des bibliothèques de sections efficaces simplifiées, n'utilisant que quelques groupes seulement. Le code APOLLO2 ([Loubière *et al.*, 1999]) est utilisé au CEA pour réaliser ces calculs de transport.
- Au niveau le plus macroscopique, un cœur a une structure hétérogène due aux différences entre les assemblages qu'on y place. Un calcul de cœur utilise un modèle neutronique simple. Par exemple, un calcul standard de réacteur à eau sous pression (présenté section 1.1.1) utilise une approximation de la diffusion à deux groupes d'énergie. Si une plus grande précision est recherchée, le modèle du transport simplifié ($SP_N$, voir section 1.2.3) peut-être utilisé, avec éventuellement un nombre de groupes d'énergie plus élevé. Les calculs de diffusion et $SP_N$ sont assurés au CEA par le code CRONOS2 ([Baudron *et al.*, 1999]).

Notons que d'autres étapes sont nécessaires, telles que le calcul du réflecteur, la prise en compte de l'évolution et des contre-réactions, la cinétique rapide...

Un des enjeux actuel est de pouvoir réaliser des calculs de cœur à trois dimensions d'espace en prenant en compte la structure spatiale fine au niveau de la cellule, avec un temps de calcul acceptable pour des industriels. A l'avenir, l'augmentation de la puissance de calcul des ordinateurs et l'amélioration des méthodes numériques permettront sans doute de réaliser des

calculs de transport sur le cœur complet, en deux voir trois dimensions d'espace.

---

*Après avoir introduit le fonctionnement d'un réacteur nucléaire, en particulier les réacteurs à eau sous pression (REP), nous avons rappelé les équations de la neutronique. Les deux principales sont l'équation du transport des neutrons, et l'équation de la diffusion des neutrons. Nous avons ensuite présenté les méthodes numériques pour résoudre ces équations, et les schémas de calcul qui permettent d'étudier avec précision la population des neutrons dans un cœur de réacteur nucléaire.*

*Nous allons nous focaliser dans le chapitre suivant sur un solveur de cœur : le solveur MINOS, qui résout l'équation de la diffusion et les équations $SP_N$ par une méthode d'éléments finis mixte duale.*

---

# Chapitre 2

# Présentation du solveur de cœur MINOS

---

*Au chapitre précédent, nous avons introduit la neutronique d'un cœur de réacteur nucléaire, et nous avons présenté les différentes méthodes numériques permettant de réaliser un schéma de calcul complet pour modéliser la population de neutrons. Le calcul de cœur est la dernière maille de cette chaîne, et nous allons présenter dans ce chapitre un solveur de cœur développé au CEA : le solveur MINOS. Ce solveur permet de résoudre l'équation de la diffusion et les équations $SP_N$, introduites au chapitre précédent. Afin de rendre l'exposé plus clair, nous nous limiterons à la présentation de la résolution de l'équation de la diffusion, mais l'extension aux équations $SP_N$ n'est pas complexe, puisqu'elles se ramènent en fait à des équations de diffusion couplées. De même, nous ne considérerons que l'équation monocinétique, mais là aussi l'extension à plusieurs groupes d'énergie ne pose pas de problème, car la résolution est faite sur chaque groupe avec un couplage sur les termes sources.*

*Le solveur MINOS utilise une méthode d'éléments finis, basée sur une formulation variationnelle de l'équation de la diffusion. Dans un premier temps nous introduirons donc les différentes formulations variationnelles de l'équation de la diffusion, puis nous étudierons les méthodes numériques utilisées par le solveur MINOS. Après l'étude du solveur, nous nous focaliserons sur l'intégration de MINOS dans le projet APOLLO3/DESCARTES, en décrivant les classes C++ utilisées. Enfin les géométries DESCARTES des cœurs du REP et du RJH sont présentées en fin de chapitre.*

---

Nous avons introduit l'équation monocinétique de la diffusion des neutrons (1.10) dans le cas d'un calcul critique au chapitre précédent. Afin de rendre la formulation plus synthétique, nous adoptons dorénavant l'écriture suivante :

$$-\overrightarrow{\nabla}.(D\overrightarrow{\nabla}\varphi)+\sigma_a\varphi=\frac{1}{k_{eff}}\sigma_f\varphi, \tag{2.1}$$

avec :

- D le coefficient de diffusion ;
- $\sigma_a$ la section efficace d'absorbtion ;
- $\sigma_f$ la section efficace de fission multipliée par le nombre moyen de neutrons émis par fission ;
- $\varphi$ le flux de neutrons ;
- $k_{eff}$ le facteur de multiplication effectif.

Avec l'ajout de conditions aux bords qui permettent de fermer le problème, l'équation (2.1) admet un seul vecteur propre $\varphi$ ne changeant pas de signe, qui correspond à la plus grande valeur propre $k_{eff}$ ([Planchard, 1995], [Dautray et Lions, 1988]). C'est la solution "physique" du problème, puisque le flux de neutrons $\varphi$ est par nature positif.

Le solveur MINOS utilise une formulation mixte : le courant que nous notons maintenant $\overrightarrow{p}$, défini à l'équation (1.6) par $\overrightarrow{p}=-D\overrightarrow{\nabla}\varphi$, devient une inconnue auxiliaire. L'équation (2.1) est ainsi équivalente au système suivant :

$$\begin{cases} \overrightarrow{p}+D\overrightarrow{\nabla}\varphi=0, \\ \overrightarrow{\nabla}.\overrightarrow{p}+\sigma_a\varphi=\dfrac{1}{k_{eff}}\sigma_f\varphi. \end{cases} \tag{2.2}$$

Nous notons dorénavant $R$ le domaine de calcul, c'est à dire le cœur du réacteur. De même que pour l'équation (2.1), le système d'équations (2.2) n'est pas fermé : il est nécessaire d'ajouter des conditions sur le bord du cœur $\partial R$. Plusieurs types de conditions aux bords sont utilisés en neutronique, notamment :

- des conditions de flux nul : $\varphi=0$ sur $\partial R$ ;
- des conditions de courant nul, appelées également conditions de réflexion ou de milieu infini : $\overrightarrow{p}=0$ sur $\partial R$. Cette condition est équivalente à une condition de Neumann : $\partial\varphi/\partial n=0$, où $\overrightarrow{n}$ est la normale sortante ;
- des conditions mixtes de Robin (ou d'albedo) : $D\partial\varphi/\partial n+\alpha\varphi=0$ sur $\partial R$, qui peuvent se réecrire avec le courant : $-\overrightarrow{p}\cdot\overrightarrow{n}+\alpha\varphi=0$. Le coefficient $\alpha$ est positif et dépend du matériau et du problème traité.

Nous utilisons dorénavant des conditions de flux nul aux bords, sachant que les autres types de conditions ne modifient pas les méthodes que nous allons présenter. Le problème à résoudre est donc de trouver $(\overrightarrow{p},\varphi,k_{eff})$ solution fondamentale de :

$$\begin{cases} \overrightarrow{p}+D\overrightarrow{\nabla}\varphi=0 \quad \text{sur } R, \\ \overrightarrow{\nabla}.\overrightarrow{p}+\sigma_a\varphi=\dfrac{1}{k_{eff}}\sigma_f\varphi \quad \text{sur } R, \\ \varphi=0 \quad \text{sur } \partial R. \end{cases} \tag{2.3}$$

## 2.1 Les formulations variationnelles de l'équation de la diffusion

### 2.1.1 Les espaces de Sobolev

Nous commençons par rappeler la définition de certains espaces de Sobolev, nécessaires pour définir les formulations variationnelles du problème (2.3) :

$$
\begin{aligned}
&L^2(R) = \left\{ \psi : R \to \mathbb{R}, \int_R |\psi(x)|^2 \, dx < +\infty \right\} ; \\
&H^1(R) = \left\{ \psi \in L^2(R), \frac{\partial \psi}{\partial x_d} \in L^2(R), 1 \leq d \leq N_d \right\} ; \\
&H^1_{0,\Gamma}(R) = \left\{ \psi \in H^1(R), \psi_{|\Gamma} = 0 \right\} \text{ avec } \Gamma \subset \partial R ; \\
&H(div, R) = \left\{ \overrightarrow{q} \in (L^2(R))^{N_d}, \overrightarrow{\nabla} . \overrightarrow{q} \in L^2(R) \right\} ; \\
&H_{0,\partial R}(div, R) = \{ \overrightarrow{q} \in H(div, R), \overrightarrow{q} \cdot \overrightarrow{n} = 0 \text{ sur } \partial R \} ;
\end{aligned}
\tag{2.4}
$$

avec l'espace produit $(L^2(R))^{N_d} = \left\{ \overrightarrow{q} = (q_d)_{1 \leq d \leq N_d}, q_d \in L^2(R), 1 \leq d \leq N_d \right\}$. $N_d$ est la dimension de l'espace. On définit $H^1_0(R) = H^1_{0,\partial R}(R)$. Les normes associées à ces différents espaces sont les suivantes :

$$
\begin{aligned}
\|\psi\|_{L^2(R)} &= \left( \int_R |\psi(x)|^2 \, dx \right)^{\frac{1}{2}} ; \\
\|\psi\|_{H^1(R)} &= \left( \|\psi\|_{L^2(R)} + \sum_{d=1}^{N_d} \left\| \frac{\partial \psi}{\partial x_d} \right\|^2_{L^2(R)} \right)^{\frac{1}{2}} ; \\
\|\psi\|_{H(div,R)} &= \left( \|\overrightarrow{q}\|^2_{L^2(R)} + \left\| \overrightarrow{\nabla} . \overrightarrow{q} \right\|^2_{L^2(R)} \right)^{\frac{1}{2}} .
\end{aligned}
\tag{2.5}
$$

Le solveur MINOS utilise une méthode d'éléments finis pour résoudre le problème (2.3). Il est donc nécessaire de le mettre sous forme variationnelle. La méthodologie pour obtenir cette formulation est la suivante :

- multiplication du problème "fort" par une fonction "test" appartenant à un espace du même type que celui de la solution ;
- intégration des équations sur tout le domaine ;
- application éventuelle de la formule de Green afin d'avoir le même degré de différenciation sur la solution et sur la fonction "test".

Cette formulation permet notamment de travailler dans des espaces plus "faibles" que ceux imposés par le problème "fort", c'est à dire moins contraignants. Nous renvoyons à l'ouvrage de référence [Brézis, 1983] pour toutes précisions sur les espaces de Sobolev et les formulations variationnelles d'équations aux dérivées partielles.

### 2.1.2 Les formulations variationnelles mixtes primale et duale

La formulation mixte primale s'obtient en multipliant la première équation de (2.3) par une fonction test de courant et la deuxième équation par une fonction test de flux. Puis on intègre les deux équations sur le domaine de calcul $R$, et on applique la formule de Green au premier terme du membre de gauche de la deuxième équation. Le problème est alors de trouver la solution fondamentale $\varphi \in H^1_0(R)$, $\overrightarrow{p} \in (L^2(R))^{N_d}$, $k_{eff} \in \mathbb{R}$ du système d'équations :

$$
\left\{
\begin{array}{l}
\displaystyle \int_R \frac{1}{D} \overrightarrow{p} . \overrightarrow{q} + \int_R \overrightarrow{\nabla} \varphi . \overrightarrow{q} = 0 \quad \forall \overrightarrow{q} \in (L^2(R))^{N_d}, \\
\displaystyle - \int_R \overrightarrow{p} . \overrightarrow{\nabla} \psi + \int_R \sigma_a \varphi \psi = \frac{1}{k_{eff}} \int_R \sigma_f \varphi \psi \quad \forall \psi \in H^1_0(R).
\end{array}
\right.
\tag{2.6}
$$

Pour obtenir la formulation mixte duale, il faut appliquer la formule de Green sur un autre terme : le deuxième terme du membre de gauche de la première équation. Le problème est

alors de trouver la solution fondamentale $\varphi \in L^2(R), \overrightarrow{p} \in H(div, R), k_{eff} \in \mathbb{R}$ du système d'équations :

$$\begin{cases} \int_R -\frac{1}{D}\overrightarrow{p}.\overrightarrow{q} + \int_R \overrightarrow{\nabla}.\overrightarrow{q}\,\varphi = 0 \quad \forall \overrightarrow{q} \in H(div, R), \\ \int_R \overrightarrow{\nabla}.\overrightarrow{p}\,\psi + \int_R \sigma_a \varphi\psi = \frac{1}{k_{eff}} \int_R \sigma_f \varphi\psi \quad \forall \psi \in L^2(R). \end{cases} \tag{2.7}$$

C'est ce problème sous cette formulation qui est résolu par le solveur MINOS. Nous l'utiliserons également dans notre travail.

Une fois prouvées l'existence et l'unicité de la solution de (2.7), il n'est pas difficile de montrer l'équivalence entre le problème fort (2.3) et le problème faible (2.7). En supposant que le flux $\varphi$ est suffisamment régulier, le couple $(\varphi, k_{eff})$ est solution de (2.3) si et seulement si il est solution de (2.7).

### 2.1.3 Existence et unicité de la solution du problème mixte duale

Nous avons vu que le problème fort de la diffusion (équation (2.3)) admet une solution unique telle que le flux ne change pas de signe sur $R$. Il nous faut maintenant montrer que la solution du problème mixte duale (2.7) existe et est unique. Pour cela, nous considérons le problème à source suivant : trouver $\varphi \in L^2(R), \overrightarrow{p} \in H(div, R)$ solution de

$$\begin{cases} \int_R -\frac{1}{D}\overrightarrow{p}.\overrightarrow{q} + \int_R \overrightarrow{\nabla}.\overrightarrow{q}\,\varphi = 0 \quad \forall \overrightarrow{q} \in H(div, R), \\ \int_R \overrightarrow{\nabla}.\overrightarrow{p}\,\psi + \int_R \sigma_a \varphi\psi = \int_R S\psi \quad \forall \psi \in L^2(R). \end{cases} \tag{2.8}$$

De façon à nous placer dans un cadre plus abstrait, nous réecrivons ce système sous la forme : trouver $\varphi \in V, p \in W$ solution de

$$\begin{cases} a(p,q) + b(q,\varphi) = f(q) \quad \forall q \in W, \\ b(p,\psi) - t(\varphi,\psi) = g(\psi) \quad \forall \psi \in V. \end{cases} \tag{2.9}$$

La correspondance avec (2.8) est :

$$\begin{aligned} & W = H(div, R), V = L^2(R); \\ & a(\overrightarrow{p}, \overrightarrow{q}) = \int_R \frac{1}{D}\overrightarrow{p}.\overrightarrow{q}; \\ & b(\overrightarrow{p}, \psi) = -\int_R \overrightarrow{\nabla}.\overrightarrow{p}\,\psi; \\ & t(\varphi,\psi) = \int_R \sigma_a\varphi\psi; \\ & g(\psi) = -\int_R S\psi; \\ & f \equiv 0; \end{aligned} \tag{2.10}$$

avec $a, b, t$ qui sont des formes bilinéaires continues.

Nous renvoyons à [Brezzi et Fortin, 1991] et [Roberts et Thomas, 1991] pour les démonstrations des théorèmes 2.1 et 2.2 d'existence et d'unicité qui suivent. Rappelons qu'une forme bilinéaire $a$ est $W$-elliptique si $\exists \alpha > 0, a(q,q) \geq \alpha \|q\|_W^2, \forall q \in W$.

**Théorème 2.1** *Si $a$ est $W$-elliptique et $t$ est $V$-elliptique, alors le problème (2.9) admet une solution unique, $\forall f \in W'$ (dual de $W$), $\forall g \in V'$.*

La preuve est simplement une application du théorème de Lax-Milgram au problème obtenu en faisant la différence des deux équations du problème (2.9). En effet, la forme bilinéaire $k((p,\varphi),(q,\psi)) = a(p,q) + b(q,\varphi) + t(\varphi,\psi) - b(p,\psi)$ est elliptique, et le second membre $F(q,\psi) = f(q) - g(\psi)$ est linéaire.

Mais ce théorème n'est pas applicable au problème (2.8), car la forme $a$ n'est pas elliptique sur $H(div,R)$ pour la norme $\|.\|_{H(div,R)}$. Nous avons donc besoin de conditions moins restrictives sur $a$, mais en contrepartie il va nous falloir introduire des hypothèses sur la forme $b$.

**Théorème 2.2** *Soit l'espace* $KerB = \{q \in W, b(q,\psi) = 0, \forall \psi \in V\}$. *On suppose que :*
- *$a$ est semi-définie positive et (KerB)-elliptique :* $\exists \alpha > 0, a(q,q) \geq \alpha \|q\|^2_W, \forall q \in KerB$ ;
- *$b$ vérifie la condition inf-sup :* $\exists \beta > 0, \inf_{\psi \in V} \sup_{q \in W} \frac{b(q,\psi)}{\|q\|_W \|\psi\|_V} \geq \beta > 0$ ;
- *$t$ est semi-définie positive symétrique ;*

*alors le problème (2.9) admet une solution unique,* $\forall f \in W'$, $\forall g \in V'$.

Les hypothèses de ce théorème sont vérifiées par le problème (2.8). On renvoie à la thèse [Schneider, 2000] pour plus de détails à ce sujet.

### 2.1.4 Approximation de la solution par une méthode de Galerkin

Le problème (2.9) n'ayant pas de solution analytique, sauf dans des cas particuliers simples, des méthodes de résolution numérique sont utilisées afin d'obtenir une approximation de la solution exacte. Tout l'enjeu d'une méthode numérique performante est d'obtenir la meilleure approximation possible, avec un "coût" algorithmique le plus faible possible. De nombreuses méthodes numériques de résolution des équations aux dérivées partielles existent, permettant de résoudre des problèmes différents. Nous nous plaçons dans le cadre de problèmes elliptiques (comme le problème (2.9)), et nous allons nous intéresser par la suite à des méthodes variationnelles de type Galerkin (la méthode des éléments finis en est un exemple). Ces méthodes introduisent des espaces de dimensions finies, de manière à chercher une projection de la solution du problème sur ces espaces selon les produits scalaires définis naturellement par la formulation variationnelle du problème. Dans notre cas, ces produits scalaires sont les formes bilinéaires $a$ et $t$ définies par l'équation (2.10).

Dans le cadre de notre problème abstrait (2.9), nous introduisons donc deux espaces $W_h$ et $V_h$ de dimensions finies. Nous considérons une approximation conforme : $W_h \subset W$ et $V_h \subset V$. Le problème discret à résoudre est alors le suivant : trouver $\varphi_h \in V_h$ et $p_h \in W_h$ tels que :

$$\begin{cases} a(p_h,q_h) + b(q_h,\varphi_h) = f(q_h) & \forall q_h \in W_h, \\ b(p_h,\psi_h) - t(\varphi_h,\psi_h) = g(\psi_h) & \forall \psi_h \in V_h. \end{cases} \tag{2.11}$$

Soit l'espace $KerB_h = \{q_h \in W_h, b(q_h,\psi_h) = 0, \forall \psi_h \in V_h\}$. Sous les hypothèses suivantes :

$$\exists \alpha_h > 0, a(q_h,q_h) \geq \alpha_h \|q_h\|^2_{W_h}, \forall q_h \in KerB_h, \tag{2.12}$$

$$\exists \beta_h > 0, \inf_{\psi_h \in V_h} \sup_{q_h \in W_h} \frac{b(q_h,\psi_h)}{\|q_h\|_{W_h} \|\psi_h\|_{V_h}} \geq \beta_h > 0, \tag{2.13}$$

le théorème 2.2 s'applique et garantit que le problème (2.11) a une solution unique.

Il est important de maîtriser l'erreur d'une méthode d'approximation, c'est à dire de connaître une borne supérieure de la différence entre la solution du problème discret (2.11), et celle du problème "approché" (2.9). C'est l'objet du théorème suivant :

**Théorème 2.3** *On suppose que :*
- *a est semi-définie positive, (KerB)-elliptique et uniformément ($KerB_h$)-elliptique ($\alpha_0 = \inf_h \alpha_h > 0$) ;*
- *t est semi-définie positive et symétrique ;*
- *b vérifie la condition inf-sup discrète uniforme ($\beta_0 = \inf_h \beta_h > 0$).*

*Alors il existe une constante C (dépendante des formes a,b,t) telle que si $(\varphi, p)$ est solution de (2.9) et si $(\varphi_h, p_h)$ est solution de (2.11), alors :*

$$\|p - p_h\|_W + \|\varphi - \varphi_h\|_V \leq C \left( \inf_{q_h \in W_h} \|p - q_h\|_W + \inf_{\psi_h \in V_h} \|\varphi - \psi_h\|_V \right). \tag{2.14}$$

La résolution du problème discret (2.11) correspond à une projection de la solution exacte sur les espaces $W_h$ et $V_h$, et le théorème 2.3 garantit par (2.14) une optimalité de la solution approchée dans ces espaces discrets. En pratique, la résolution du problème discret mène à la résolution d'un système linéaire. De nombreuses méthodes permettent de résoudre ce système : méthodes de type gradient conjugué([Saad, 2003]), méthodes de factorisation de type "LU" ([Ciarlet, 1982])... La rapidité de la résolution du problème discret dépend donc entre autres de l'efficacité de ces méthodes.

L'application du théorème 2.3 au problème (2.8) nécessite, en plus de la condition inf-sup discrète uniforme, de montrer la $(KerB_h)$-ellipticité de la forme $a$. La proposition suivante est une condition suffisante pour la vérifier :

**Proposition 2.1** *Soit la condition sur les espaces discrets $W_h$ et $V_h$ :*

$$\overrightarrow{q_h} \in W_h \ \text{et} \ \int_R \overrightarrow{\nabla}.\overrightarrow{q_h}\psi_h = 0, \forall \psi_h \in V_h \Rightarrow \overrightarrow{\nabla}.\overrightarrow{q_h} = 0. \tag{2.15}$$

*Dans le cadre du problème (2.8), si les sous-espaces $W_h \subset H(div, R)$ et $V_h \subset L^2(R)$ vérifient la condition (2.15), on a $KerB_h \subset KerB$, ce qui implique la $(KerB_h)$-ellipticité uniforme de la forme a.*

On renvoie à [Roberts et Thomas, 1991] pour la démonstration de cette proposition.

## 2.2 Les méthodes numériques du solveur MINOS

Le solveur MINOS est un solveur pour des calculs de cœur , qui constituent, comme nous l'avons vu à la section 1.2.5, la dernière étape d'un schéma de calcul de la neutronique d'un réacteur. Ce solveur utilise une méthode d'éléments finis dite mixte duale, en référence à la formulation (2.7) utilisée pour résoudre l'équation de la diffusion ou les équations $SP_N$.

Sans rentrer dans les détails, nous rappelons le principe de la méthode des éléments finis. C'est une méthode de type Galerkin : la solution approchée d'un problème variationnel est cherchée comme une projection de la solution exacte sur un espace discret de dimension finie, engendré par des fonctions de base définies sur un maillage. Ces fonctions sont obtenues par composition de quelques fonctions polynomiales sur une géométrie de référence, et d'une transformation affine faisant passer de la géométrie de référence à la maille courante. La solution approchée est alors une combinaison linéaire des différentes fonctions de base. Les coefficients de cette combinaison linéaire sont les inconnues ou degrés de liberté du problème discret, représentés par des "nœuds " du maillage.

Pour simplifier l'exposé, nous nous plaçons dorénavant en deux dimensions d'espace, mais la généralisation à trois dimensions ne pose aucun problème.

### 2.2.1 Le maillage cartésien et les espaces discrets

Le solveur MINOS travaille en géométrie rectangulaire, ce qui simplifie grandement l'élaboration du maillage. En effet un maillage en deux dimensions est construit comme le produit de deux maillages monodimensionnelles, et il suffit de préciser l'ensemble des tailles de maille dans chaque direction pour avoir entièrement défini le maillage. Nous verrons que l'efficacité du solveur MINOS provient en grande partie de cette décomposition spatiale, car elle permet de se ramener à des résolutions à une dimension dans chaque direction. En contrepartie, les géométries complexes sont souvent mal décrites par un maillage cartésien, et il sera nécessaire pour de telles géométries d'utiliser un maillage très fin. Nous noterons dorénavant $T_h$ l'ensemble des rectangles formant le maillage du domaine. L'indice $h$ correspond au diamètre (pour un rectangle la diagonale) de la plus grande maille de $T_h$. On rappelle que :

$$\begin{array}{l} \bar{R} = \bigcup_{K \in T_h} K; \\ \forall K_1 \in T_h, \forall K_2 \in T_h, \mathring{K_1} \cap \mathring{K_2} = \emptyset. \end{array} \tag{2.16}$$

Les espaces discrets utilisés par MINOS sont inclus dans leurs homologues de dimensions infinies, de façon à avoir une approximation conforme. Soit $P(K)$ l'ensemble des polynômes de degré quelconque sur $K$. En dimension 2, on désignera par $Q_{l,m}(K)$ l'espace des polynômes de degré inférieur à $l$ en $x$ et $m$ en $y$ :

$$Q_{l,m}(K) = \left\{ p(x,y) \in P(K), p(x,y) = \sum_{i,j=0}^{l,m} a_{ij} x^i y^j, \quad a_{ij} \in \mathbb{R} \right\}. \tag{2.17}$$

Nous introduisons également l'espace de polynômes $D_k(K)$ :

$$D_k(K) = \left[ Q_{k,k-1}(K) \times \{0\} \right] \oplus \left[ \{0\} \times Q_{k-1,k}(K) \right]. \tag{2.18}$$

On définit alors l'espace d'approximation pour le flux :

$$V_h^k(R) = \left\{ \psi \in L^2(R), \forall K \in T_h, \psi_{|K} \in Q_{k-1,k-1}(K) \right\}. \tag{2.19}$$

Pour le courant, l'espace d'approximation est :

$$W_h^k(R) = \left\{ \overrightarrow{q} \in H(div, R), \forall K \in T_h, \overrightarrow{q}_{|K} \in D_k(K) \right\}. \tag{2.20}$$

La conformité de l'approximation de courant ($W_h^k(R) \subset H(div, R)$) implique que la trace normale du courant à l'interface entre deux mailles soit continue.

Les espaces $V_h^k(R)$ et $W_h^k(R)$ vérifient la condition $inf - sup$ uniforme du théorème 2.3 et la condition (2.15) (voir [Roberts et Thomas, 1991] pour la démonstration). Les hypothèses du théorème 2.3 sont donc vérifiées.

### 2.2.2 Le système linéaire

Soient $(\varphi_i)_{1 \leq i \leq n}$ un ensemble de fonctions de base de flux, et $(\overrightarrow{p_i})_{1 \leq i \leq m}$ un ensemble de fonctions de base de courant, tels que :

$$V_h^k(R) = Vect\{\varphi_i\}_{1 \leq i \leq n} \quad \text{et} \quad W_h^k(R) = Vect\{\overrightarrow{p_i}\}_{1 \leq i \leq n}. \tag{2.21}$$

La résolution du problème discret (2.11) est équivalente à la résolution d'un système linéaire : les matrices et les vecteurs du second membre sont obtenus par application des formes bilinéaires $a, b, t$ et des seconds membres linéaires $f, g$ sur l'ensemble des fonctions de base. Les

inconnues du système sont les vecteurs composés des degrés de liberté de flux et de courant. Le système linéaire correspondant au problème discret (2.11) a la forme suivante, en supposant que la source de la première équation est nulle ($f \equiv 0$) :

$$\begin{cases} -AP + B\Phi = 0, \\ B^T P + T\Phi = S. \end{cases} \tag{2.22}$$

$\Phi, P$ sont les vecteurs des inconnues de flux et de courant respectivement. Les matrices et le second membre sont définis comme suit :

$$\begin{aligned} A(i,j) &= a(\overrightarrow{p_i}, \overrightarrow{p_j}) = \int_R \frac{1}{D} \overrightarrow{p_i}.\overrightarrow{p_j}; \\ B(i,j) &= -b(\overrightarrow{p_i}, \varphi_j) = \int_R \overrightarrow{\nabla}.\overrightarrow{p_i}\varphi_j; \\ T(i,j) &= t(\varphi_i, \varphi_j) = \int_R \sigma_a \varphi_i \varphi_j; \\ S(i) &= -g(\varphi_i) = \int_R S\varphi_i. \end{aligned} \tag{2.23}$$

En fait l'inconnue de courant est décomposée en ses différentes composantes dans chaque direction. De plus, nous allons voir que les fonctions de base de courant utilisées n'ont qu'une seule composante non nulle. Le système (2.22) résolu par MINOS se réecrit alors :

$$\begin{cases} -A_d P_d + B_d \Phi = 0 \quad 1 \geq d \geq N_d, \\ \sum_{d=1}^{N_d} B_d^T P_d + T\Phi = S, \end{cases} \tag{2.24}$$

avec :

$$\begin{aligned} A_d(i,j) &= a_d(\overrightarrow{p_i}, \overrightarrow{p_j}) = \int_R \frac{1}{D} p_{i,d} p_{j,d}, \\ B_d(i,j) &= -b_d(\overrightarrow{p_i}, \varphi_j) = \int_R \frac{\partial p_{i,d}}{\partial d} \varphi_j. \end{aligned} \tag{2.25}$$

L'indice $d$ représente une des $N_d$ directions de l'espace, et $p_{i,d}$ est la composante de $\overrightarrow{p_i}$ dans la direction $d$. La formulation globale de ce système, avec $P = [P_x, P_y]^T$ en 2D, est :

$$\begin{bmatrix} -A_x & 0 & B_x \\ 0 & -A_y & B_y \\ B_x^T & B_y^T & T \end{bmatrix} \begin{bmatrix} P_x \\ P_y \\ \Phi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ S \end{bmatrix}. \tag{2.26}$$

Nous allons voir que les coefficients des matrices sont déterminés à partir de matrices dites de référence, obtenues sur une géométrie de référence avec l'élément de Raviart-Thomas. Une transformation affine faisant passer de la géométrie de référence à la maille courante permet ensuite de déterminer les matrices complètes à partir de ces matrices de référence.

### 2.2.3 L'élément de Raviart-Thomas

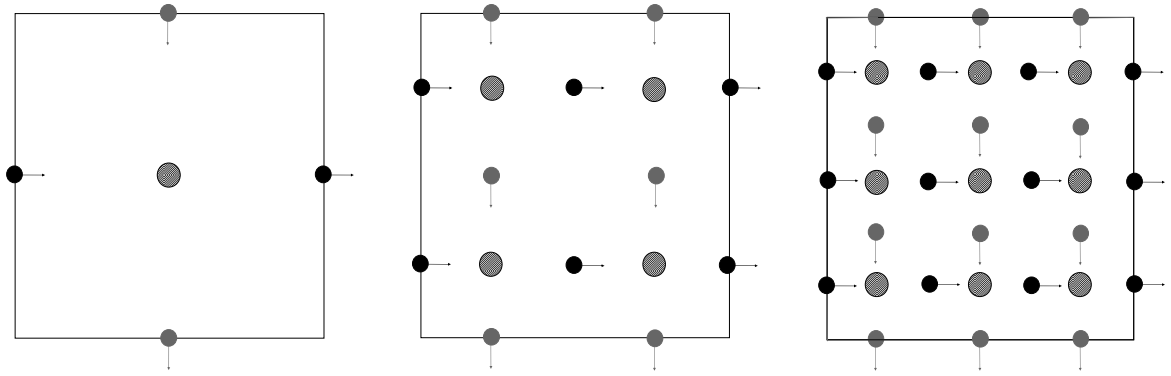
Le solveur MINOS utilise l'élément fini mixte de Raviart-Thomas en géométrie rectangulaire (introduit par [Raviart et Thomas, 1977] en 2D et [Nedelec, 1986a] en 3D). Cette élément est noté $RT_k$, k faisant référence au degré des polynômes de base. La géométrie de référence $\hat{K}$ est le cube unité en dimension 3, ou le carré unité en dimension 2. Nous utilisons le symbole "$\hat{}$" dès que nous nous plaçons sur l'élément de référence.

Sur la géométrie de référence, l'élément $RT_{k-1}$ utilise les espaces $Q_{k-1,k-1}(\hat{K})$ pour le flux et $D_k(\hat{K})$ pour le courant (définis par (2.17) et (2.18)). La composante $d$ du courant est donc dans un espace de polynômes de degré $k$ pour la variable correspondant à la direction $d$, et $k-1$ pour les autres directions. On a donc la propriété suivante, qui lie les espaces discrets de flux et de courant : si $\overrightarrow{f} \in D_k(\hat{K})$, $\overrightarrow{\nabla}.\overrightarrow{f} \in Q_{k-1,k-1}(\hat{K})$. En d'autres termes, la divergence d'un polynôme de l'espace discret de courant appartient à l'espace discret de flux. La dimension de $Q_{k-1,k-1}(\hat{K})$ est $k^2$, celle de $D_k(\hat{K})$ est $2k(k+1)$.

La figure 2.1 représente les trois premiers éléments de Raviart-Thomas sur le carré unité :

- $RT_0$ (linéaire, figure 2.1a), $\overrightarrow{p} \in Q_{10} \times Q_{01}, \varphi \in Q_{00}$ ;
- $RT_1$ (parabolique, figure 2.1b), $\overrightarrow{p} \in Q_{21} \times Q_{12}, \varphi \in Q_{11}$ ;
- $RT_2$ (cubique, figure 2.1c), $\overrightarrow{p} \in Q_{32} \times Q_{23}, \varphi \in Q_{22}$.

Les nœuds de flux sont les nœuds internes barrés, tandis que ceux de courant sont les nœuds fléchés horizontalement pour la composante en $x$, verticalement pour la composante en $y$.



(a) $RT_0$ (b) $RT_1$ (c) $RT_2$

FIG. 2.1: Élément de Raviart-Thomas $RT_0$, $RT_1$ et $RT_2$.

### 2.2.4 Passage de l'élément de référence à l'élément courant

Supposons que nous travaillons avec un maillage rectangulaire régulier, c'est à dire que toutes les mailles sont des rectangles de même dimension $(h_x, h_y)$. L'application $F$ qui permet de passer de l'élément de référence $\hat{K} = [0,1]^2$ à l'élément $K = [x_K, x_K + h_x] \times [y_K, y_K + h_y]$ est définie par :

$$F \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} = \begin{pmatrix} h_x \hat{x} \\ h_y \hat{y} \end{pmatrix} + \begin{pmatrix} x_K \\ y_K \end{pmatrix}. \tag{2.27}$$

Le Jacobien de cette transformation est $J = h_x h_y$. Les formes bilinéaires (2.10) peuvent être écrites comme une somme sur toutes les mailles d'intégrales exprimées sur l'élément de référence après un changement de variable. Si l'on note $p_d$ la composante dans la direction $d$

du courant $\overrightarrow{p}$, on obtient :

$$\begin{aligned}
a_d(\overrightarrow{p},\overrightarrow{q}) &= \sum_{K\in T_h}\frac{1}{D_K}\int_K p_d q_d = \sum_{K\in T_h}\frac{1}{D_K}\frac{h_d^2}{J}\int_{\hat{K}}\hat{p}_d\hat{q}_d;\\
b_d(\overrightarrow{q},\varphi) &= \sum_{K\in T_h}\int_K \frac{\partial q_d}{\partial d}\varphi = \sum_{K\in T_h}\int_{\hat{K}}\frac{\partial \hat{q}_d}{\partial d}\hat{\varphi};\\
t(\varphi,\psi) &= \sum_{K\in T_h}\sigma_a^K\int_K\varphi\psi = \sum_{K\in T_h}J\sigma_a^K\int_{\hat{K}}\hat{\varphi}\hat{\psi};\\
S(\psi) &= \sum_{K\in T_h}S_K\int_K\psi = \sum_{K\in T_h}S_K J\int_{\hat{K}}\hat{\psi}.
\end{aligned} \tag{2.28}$$

Les sections efficaces sont supposées constantes par maille dans le solveur MINOS. Il est donc nécessaire de les avoir auparavant homogénéisées sur chaque maille. $D_K, \sigma_a^K, S_K$ sont donc constants sur chaque maille, correspondant respectivement au coefficient de diffusion, à la section efficace d'absorption et au terme source.

### 2.2.5 Les fonctions de base pour le flux et le courant

Nous allons maintenant déterminer les polynômes qui forment les bases de flux et de courant. Le choix de cette base doit permettre de simplifier au maximum les matrices du système linéaire (2.24) à résoudre. Notamment on souhaite avoir des matrices les plus creuses possible, de façon à minimiser le coût d'inversion du système. Il est donc nécessaire de limiter les couplages entre les différents nœuds . On se place dans le cadre d'un élément $RT_{k-1}$.

#### La base de flux

La base de flux est choisie de manière à ce que la matrice $T$ de couplage des flux soit diagonale : un polynôme de la base de flux ne doit être couplé avec aucun autre. De plus, afin de découpler les calculs par direction, ces polynômes sont choisis comme le produit de polynômes 1D :

$$\hat{\varphi}_i(\hat{x},\hat{y}) = L_{i_x}(\hat{x})L_{i_y}(\hat{y}), \tag{2.29}$$

où $1 \le i_x, i_y \le k$ et $i = (i_y-1)k + i_x$. Il y a donc $k^2$ polynômes de base de flux pour l'élément $RT_{k-1}$ en 2D. Les $L_i$ sont des polynômes de Lagrange de degré $k-1$ :

$$L_i(\hat{x}) = \frac{\prod_{j=1,j\neq i}^{k}(\hat{x}-\hat{x}_j)}{\prod_{j=1,j\neq i}^{k}(\hat{x}_i-\hat{x}_j)}, 1\le i\le k. \tag{2.30}$$

On a donc : $L_i(\hat{x}_j) = \delta_{ij}$. En prenant pour les $(\hat{x}_i)_{1\le i\le k}$ les points de Gauss-Legendre d'ordre k, comme la formule d'intégration de Gauss-Legendre est exacte pour un polynôme de degré $2k-1$, les $L_i$ vérifient les propriétés suivantes :

$$\begin{aligned}
&\int_0^1 L_i(\hat{x})L_j(\hat{x})d\hat{x} = t_i\delta_{ij};\\
&\sum_{i=1}^{k} L_i(\hat{x}) = 1, \forall\hat{x}\in[0,1];\\
&\int_0^1 L_i^2(\hat{x})d\hat{x} = \int_0^1 L_i(\hat{x})d\hat{x} = t_i.
\end{aligned} \tag{2.31}$$

La matrice de référence de couplage des flux $\hat{T}$, de taille $k^2$, définie par $\hat{T}(i,j) = \int_0^1 L_i(\hat{x})L_j(\hat{x})d\hat{x}$, est donc diagonale :

$$\hat{T} = \begin{pmatrix} t_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & t_k \end{pmatrix}. \tag{2.32}$$

Le développement du flux sur la base que nous venons de décrire s'écrit :

$$\hat{\varphi}(\hat{x},\hat{y}) = \sum_{i_y=1}^{k}\sum_{i_x=1}^{k} \hat{\varphi}_i L_{i_x}(\hat{x})L_{i_y}(\hat{y}). \tag{2.33}$$

Les $(\hat{\varphi}_i)_{1\leq i\leq k^2}$ sont les degrés de liberté correspondant aux valeurs du flux aux points de Gauss-Legendre. En effet, si on note $(\hat{x}_l,\hat{y}_m)_{1\leq l,m\leq k}$ ces points, on a : $\hat{\varphi}(\hat{x}_l,\hat{y}_m) = \hat{\varphi}_{(m-1)k+l}$.

**La base de courant**

Là aussi, les fonctions de base de courant sont le produit de polynômes 1D. Elles n'ont qu'une seule composante non nulle dans une direction donnée, et la factorisation pour cette composante (par exemple $x$) est de la forme :

$$\hat{p}_i^x(\hat{x},\hat{y}) = P_{i_x}(\hat{x})L_{i_y}(\hat{y}), \tag{2.34}$$

où $1 \leq i_x \leq k+1, 1 \leq i_y \leq k$ et $i = (i_y-1)(k+1)+i_x$. Il y a donc $k(k+1)$ polynômes de base de courant pour l'élément $RT_{k-1}$ en 2D. On remarque que les polynômes de Lagrange (2.30) sont utilisés dans la direction transverse $y$, ce qui évite tout couplage dans cette direction, de la même manière que pour le flux. Les polynômes $P_i$ sont choisis notamment pour avoir une matrice de couplage flux-courant la plus simple possible. On pose :

$$\begin{aligned} P_i(\hat{x}) &= \frac{1}{t_{i-1}}\int_0^{\hat{x}} L_{i-1}(t)dt - \frac{1}{t_i}\int_0^{\hat{x}} L_i(t)dt, \quad i = 2,...,k; \\ P_1(\hat{x}) &= 1 - \frac{1}{t_1}\int_0^{\hat{x}} L_1(t)dt; \\ P_{k+1}(\hat{x}) &= \frac{1}{t_k}\int_0^{\hat{x}} L_k(t)dt. \end{aligned} \tag{2.35}$$

Les propriétés des polynômes de Lagrange nous permettent de calculer les termes de la matrice de référence $\hat{B}$, de taille $(k+1)k$. Ainsi, $\hat{B}(i,j) = \int_0^1 \frac{\partial P_i}{\partial \hat{x}}(\hat{x})L_j(\hat{x})d\hat{x}$ et :

$$\hat{B} = \begin{pmatrix} -1 & & & \\ +1 & -1 & & \\ & \ddots & \ddots & \\ & & +1 & -1 \\ & & & +1 \end{pmatrix}. \tag{2.36}$$

Quant à la matrice de référence de couplage des courants $\hat{A}$ définie par $\hat{A}(i,j) = \int_0^1 P_i(\hat{x})P_j(\hat{x})d\hat{x}$, elle est pleine et de taille $(k+1)^2$.

Le fait que $P_i(0) = \delta_{i1}$ et $P_i(1) = \delta_{ik+1}$ permet d'assurer la conformité de l'approximation de courant : $W_h^k \subset H(div, R)$. En effet, les nœuds de courant à l'interface de deux mailles étant communs, la continuité des traces normales du courant est assurée.

Le courant dans la direction $x$ se développe sur cette base de la manière suivante :

$$\hat{p}_x(\hat{x}, \hat{y}) = \sum_{i_y=1}^{k} \sum_{i_x=1}^{k+1} \hat{p}_{i,x} P_{i_x}(\hat{x}) L_{i_y}(\hat{y}). \tag{2.37}$$

Les $(\hat{p}_{i,x})_{1 \leq i \leq k(k+1)}$ sont les degrés de liberté. Contrairement aux degrés de liberté de flux, ils ne correspondent pas à des valeurs ponctuelles du courant, sauf pour $i_x = 1$ et $i_x = k+1$, auquel cas ils sont égaux à la valeur de la trace normale du courant sur les bords de l'élément dans la direction x, aux nœuds correspondants. Notons qu' il y a $k$ degrés de liberté de plus pour le courant dans une direction donnée, que pour le flux.

### 2.2.6 Calcul des matrices élémentaires

On souhaite maintenant déterminer les matrices élémentaires sur une maille de taille $(h_x, h_y)$. Pour cela, on va utiliser à la fois les matrices sur l'élément de référence déterminées à la section précédente, et l'expression des différentes formes bilinéaires associées aux matrices après le changement de variable de l'élément de référence vers l'élément courant (voir (2.28)). En restant avec la même numérotation locale que sur l'élément de référence, on obtient :

$$\begin{aligned}
\tilde{A}_x(i,j) &= \frac{1}{D_K} \int_K p_i^x p_j^x = \frac{1}{D_K} \frac{h_x^2}{J} \int_{\hat{K}} \hat{p}_i^x \hat{p}_j^x = \frac{1}{D_K} \frac{h_x}{h_y} \hat{A}(i_x, j_x) t_{i_y} \delta_{i_y j_y}; \\
\tilde{B}_x(i,j) &= \int_K \frac{\partial p_i^x}{\partial x} \hat{\varphi}_j = \int_{\hat{K}} \frac{\partial \hat{p}_i^x}{\partial \hat{x}} \varphi_j = \hat{B}_x(i_x, j_x) t_{i_y} \delta_{i_y j_y}; \\
\tilde{T}(i,i) &= \sigma_a^K \int_K \varphi_i^2 = J \sigma_a^{\hat{K}} \int_K \hat{\varphi}_i^2 = h_x h_y \sigma_a^K t_{i_x} t_{i_y}.
\end{aligned} \tag{2.38}$$

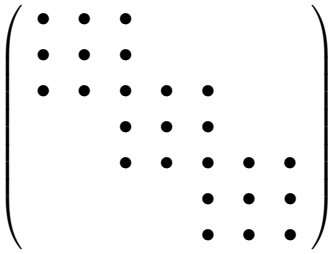
$\tilde{A}$ est une matrice pleine de dimension $(k+1)^2$, $\tilde{B}$ est bidiagonale rectangulaire de taille $k(k+1)$, et $\tilde{T}$ est diagonale de taille $k^2$.

### 2.2.7 Assemblage des matrices globales

Les nœuds de flux et de courant sont numérotés de manière canonique, en parcourant d'abord la direction $x$ puis la direction $y$. Cette numérotation, représentée figure 2.2 pour l'élément $RT_1$, permet d'optimiser le profil des matrices, et de rassembler les termes non nuls autour des diagonales.

Les matrices globales sont des matrices par blocs, les blocs étant les matrices élémentaires définies par (2.38). Après factorisation de certains termes (les tailles de maille $h_d$ et les coefficients de Lagrange $t_i$), les matrices de couplage flux-courant $B_d$ se ramènent à une matrice aux différences, de la forme de (2.36). Elles ne sont pas stockées puisque elles sont constituées uniquement de $+1$ et $-1$ répartis sur deux diagonales. La matrice de couplage des flux $T$ est diagonale et stockée sous forme d'un vecteur. Les matrices $A_d$ de couplage des courants sont constituées de blocs emboîtés de dimension $(k+1)^2$. Ils sont emboîtés car les nœuds de courant à l'interface de deux mailles sont communs, ce qui implique un couplage entre ces deux mailles. Les couplages des nœuds de courant pour l'élément $RT_1$ sont représentés sur la figure 2.3. Sur le maillage de la figure 2.2, les matrices $A_d$ sont constituées de six blocs, qui

ont chacun le profil suivant :



(2.39)

Les matrices $A_d$ et $T$ sont symétriques définies positives, car elles correspondent à des produits scalaires de fonctions de base d'espaces de dimensions finies.

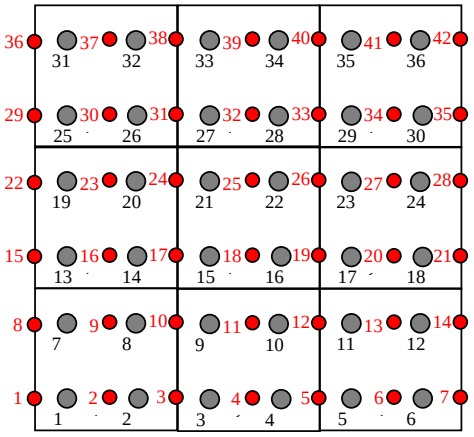



FIG. 2.2: Numérotation globale des nœuds de flux et de courant dans la direction $x$ pour l'élément $RT_1$. Les nœuds de flux sont gros et gris, ceux de courant sont petits et rouges.

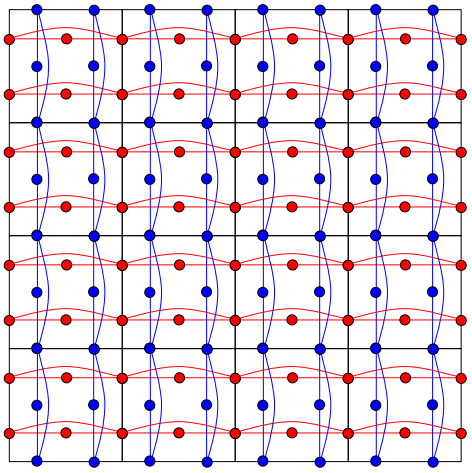


FIG. 2.3: Couplage des noeuds de courant de l'élément $RT_1$.

### 2.2.8 Résolution du système linéaire

Comme nous l'avons vu dans la section 2.2.2, le système linéaire résolu par MINOS est (ici écrit en 3D) :

$$\underbrace{\begin{bmatrix} -A_x & 0 & 0 & B_x \\ 0 & -A_y & 0 & B_y \\ 0 & 0 & -A_z & B_z \\ B_x^T & B_y^T & B_z^T & T \end{bmatrix}}_{\mathcal{H}} \begin{bmatrix} P_x \\ P_y \\ P_z \\ \Phi \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ S \end{bmatrix}. \tag{2.40}$$

Afin de rendre la matrice symétrique définie positive, l'inconnue de flux est éliminée :

$$\Phi = T^{-1}\left(S - B_x^T P_x - B_y^T P_y - B_z^T P_z\right). \tag{2.41}$$

Cette élimination est simple, car la matrice $T$ est diagonale, donc triviale à inverser, et la transposée de $B_d$ n'est pas explicitement calculée. En remplaçant le flux par son expression (2.41) dans le système (2.40), on obtient :

$$\begin{bmatrix} W_x & B_x T^{-1} B_y^T & B_x T^{-1} B_z^T \\ B_y T^{-1} B_x^T & W_y & B_y T^{-1} B_z^T \\ B_z T^{-1} B_x^T & B_z T^{-1} B_y^T & W_z \end{bmatrix} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} B_x T^{-1} S \\ B_y T^{-1} S \\ B_z T^{-1} S \end{bmatrix}. \tag{2.42}$$

avec $W_d = A_d + B_d T^{-1} B_d^T$. Le profil de la matrice $W_d$ est le même que celui de la matrice $A_d$. En effet, la matrice $T$ est diagonale (les nœuds de flux ne sont pas couplés entre eux) et les matrices $B_d$ sont composées de deux diagonales centrées, comme la matrice (2.36). Les couplages des nœuds de courant dans $W_d$ sont donc uniquement dans la direction $d$, et limités aux nœuds des mailles voisines.

La résolution du système (2.42) se fait par une méthode de Gauss-Seidel par blocs. Les blocs sont ceux représentés dans (2.42), et correspondent à une décomposition de la matrice par direction. A chaque itération de Gauss-Seidel, il est donc nécessaire de résoudre $N_d$ systèmes linéaires du type :

$$W_d P_d = B_d T^{-1} \left( S - \sum_{d' \neq d} B_{d'}^T P_{d'} \right). \tag{2.43}$$

Les matrices $W_d$ sont symétriques définies positives. En effet, $A_d$ et $T^{-1}$ le sont, et :

$$\begin{aligned} (W_d x, x) &= (A_d x, x) + (T^{-1} B_d^T x, B_d^T x) \\ &= (A_d x, x) + (T^{-1} y, y) > 0 \end{aligned} \qquad y = B_d^T x, \forall x \neq 0.$$

La méthode de Cholesky permet de résoudre le système (2.43). Bien que cette méthode soit une méthode d'inversion directe (et pas itérative), son coût est faible car le profil de la matrice $W_d$ est bon. Dans le cas de l'élément $RT_1$, dont le couplage en courant est représenté sur la figure 2.3, le profil de $W_d$ est le même que (2.39). De plus nous allons voir que la résolution d'un problème critique utilise un algorithme itératif. A chaque itération, la résolution des systèmes (2.43) est nécessaire, mais la factorisation de Cholesky est calculée une fois pour toute.

### 2.2.9 Résolution d'un problème critique à valeur propre

Comme nous l'avons vu dans la section 1.2.2, dans le cadre d'un calcul critique, les sources sont limitées aux sources de fission, et le terme source $S$ s'exprime en fonction du $k_{eff}$ et du flux :

$$S = \frac{1}{k_{eff}} \Sigma_f \Phi. \tag{2.44}$$

Le problème de diffusion critique est alors un problème à valeur propre, puisque on recherche le flux $\Phi$ et le courant $P$ associés à la plus grande valeur propre $k_{eff}$.

#### Méthode des puissances

La méthode utilisée pour résoudre ce problème à valeur propre est un algorithme itératif sur le terme source. A chaque itération, un problème à source est résolu, puis la source et la valeur propre sont mises à jour pour l'itération suivante. Le problème à résoudre s'écrit sous la forme du problème à valeur propre généralisé suivant :

$$\mathcal{H}U = \frac{1}{k} FU. \tag{2.45}$$

où $\mathcal{H}$ est la matrice introduite à l'équation (2.40), $F$ est la matrice correspondant au terme source de fission, et $U = [P_x, P_y, P_z, \Phi]$.

Supposons que l'on se donne un nombre d'itérations maximum $N^{ext}$, et un critère d'arrêt faisant intervenir une borne $\epsilon$. L'algorithme des puissances pour résoudre le problème (2.45) est le suivant :

**Algorithme 1** Algorithme des puissances

Initialisation du vecteur $U^1$ et de la valeur propre $k^1$
Calcul de la source normalisée $\bar{S}^1 = \dfrac{FU^1}{k^1}$

{Itérations externes}
**pour** $n = 1$ à $N^{ext}$ **faire**

Résolution de $\mathcal{H}U^{n+1} = \bar{S}^n$
Mise à jour de la source : $S^{n+1} = FU^{n+1}$
Calcul de la nouvelle valeur propre : $k^{n+1} = \dfrac{\left\|S^{n+1}\right\|_2^2}{(S^{n+1}, \bar{S}^n)}$
Calcul de la source normalisée : $\bar{S}^{n+1} = \dfrac{S^{n+1}}{k^{n+1}}$
Mise à jour de l'erreur : $Err = \dfrac{\left\|\bar{S}^{n+1} - \bar{S}^n\right\|_2}{\left\|\bar{S}^{n+1}\right\|_2}$
Si $Err < \epsilon$ on sort de la boucle
**fin pour**

Cet algorithme est accéléré dans MINOS par une méthode de Tchebychev sur les sources de fission ([Planchard, 1995]).

L'estimation de la convergence est en fait plus complexe que l'évaluation de $Err$. Elle porte à la fois sur la source de fission et sur la valeur propre. Il faut donc fixer une borne

supérieure $\epsilon_k$ pour le critère d'arrêt sur la valeur propre : $|k^{n+1} - k^n|$, et une borne supérieure $\epsilon_S$ pour le critère d'arrêt sur la source de fission :

$$\frac{N \times \max\limits_{1 \leq i \leq N} |\bar{S}_i^{n+1} - \bar{S}_i^n|}{\sum\limits_{i=1}^{N} |\bar{S}_i^n|} < \epsilon_S. \tag{2.46}$$

N est la taille des vecteurs sources. L'algorithme s'arrête lorsque les deux critères sont vérifiés. Des valeurs "classiques" pour un calcul de cœur sont $\epsilon_k = 10^{-5}$ et $\epsilon_S = 10^{-4}$. Le critère le plus sévère est alors dans la grande majorité des cas celui sur la source. On remarque que le critère (2.46) n'est pas défini quand la source de fission est nulle.

### L'algorithme global

Nous allons maintenant décrire l'algorithme global utilisé par le solveur MINOS pour résoudre l'équation de la diffusion critique monogroupe :

**Algorithme 2** Algorithme de MINOS

Initialisation de la source $S^0 = \Sigma_f \Phi^0$ et des fuites directionnelles $J_d^0$

{Itérations externes}
**pour** $n = 1$ à $N^{ext}$ **faire**

{Itérations internes}
**pour** $m = 1$ à $N^{int}$ **faire**

{Itérations sur les directions}
**pour** $d = 1$ à $N_d$ **faire**

Inversion du système : $W_d P_d^n = B_d T^{-1} \left[ S^{n-1} - \sum_{d'<d} J_{d'}^n - \sum_{d'>d} J_{d'}^{n-1} \right]$

Actualisation du vecteur fuite : $J_d^n = B_d^T P_d^n$

**fin pour**
**fin pour**
Mise à jour du flux $\Phi^n$
Mise à jour de la source de fission $S^n = \Sigma_f \Phi^n$
Calcul de la nouvelle valeur propre : $k_{eff}^n = \dfrac{(S^n, S^n)}{(S^n, S^{n-1})}$
Normalisation de la source : $S^n = \dfrac{S^n}{k_{eff}}$
Si les critères de convergence sur $S^n$ et $k_{eff}^n$ sont atteints, on sort de la boucle
**fin pour**

On distingue dans cette algorithme deux jeux d'itérations : les itérations internes, correspondant à la méthode de Gauss-Seidel par blocs, et les itérations externes qui assurent la convergence de la méthode des puissances pour la résolution du problème à valeur propre. Cependant, le nombre d'itérations internes $N^{int}$ est la plupart du temps fixé à 1. En effet, les

tests numériques montrent que la convergence globale assurée uniquement par les itérations externes est souvent plus rapide que lorsqu'on ajoute des itérations internes.

Le passage de l'algorithme monogroupe à l'algorithme multigroupe est simple, et nécessite l'ajout d'une boucle sur les groupes d'énergie. Un terme qui fait intervenir la section de transfert $\Sigma_s$ apparaît dans l'équation multigroupe (1.9). Le couplage entre groupe dû aux transferts et à la fission est résolu par une méthode de Gauss-Seidel à une itération : la résolution pour un groupe donné assimile les termes provenant des autres groupes à des termes sources.

## 2.3 Le projet APOLLO3/DESCARTES

Le projet APOLLO3/DESCARTES a pour ambition de créer une plateforme de développement commune à un ensemble de solveurs et de procédures permettant une large variété de calculs de neutronique. Un de ces solveurs est MINOS, entièrement revu et programmé en C++ (une version précédente est en Fortran dans le code CRONOS2, [Baudron *et al.*, 1999]).

### 2.3.1 Modèle de conception de MINOS

La programmation de MINOS en C++ a été fortement orientée objet. Le modèle de conception 2.4 met en évidence deux types de classes : des classes de base génériques de conception interne, et des classes spécifiques dérivant des classes de base pour représenter les objets que l'utilisateur construit et enchaîne afin de réaliser son calcul de façon modulaire.

Parmi les classes de conception interne du solveur, deux classes génériques représentent tous les objets de type vecteur (classe *VectorTree*) et matrice (classe *MatrixTree*) du solveur. Comme le montre la figure 2.5, ces objets sont organisés sous forme d'arbres sur les variables en énergie (groupes), en angle (harmoniques) et en espace (directions pour le courant). La classe *VectorTree* permet entre autre de représenter, sous forme unique, les discrétisations des flux pairs et impairs, et des sources. La classe *MatrixTree* est utilisée pour créer les matrices nécessaires aux différents calculs du solveur : les matrices de fission, de disparition, de transfert... Ci-dessous nous décrivons quelques classes spécifiques accessibles à l'utilisateur :

- La classe *Domain* est utilisée en entrée de MINOS. Elle permet de décrire la géométrie du domaine, ainsi que les différents milieux qui la composent.
- La classe *MacroSection*, également en entrée de MINOS, définit sur chaque région du domaine la valeur des sections efficaces macroscopiques à partir de sections microscopiques. Elle permet ainsi de stocker pour les différents milieux, les valeurs de $D, \Sigma_a, \Sigma_f$ ainsi que celles de la section de transfert $\Sigma_s$ qui apparaît dans les calculs multigroupes (voir l'équation de la diffusion multigroupe (1.9)).
- La classe *GeomCart* construit la géométrie cartésienne de MINOS. Elle peut être créée par projection à partir de la classe *Domain*. Un maillage est défini à ce stade.
- La classe *MailCart* définit le maillage spatial cartésien du calcul par la méthode des éléments finis. Le maillage de calcul peut être différent du maillage de la géométrie.
- La classe *MinosFEMGather* définit les éléments finis sur le maillage spatial. Notamment on y trouve les matrices de référence de l'élément de Raviart-Thomas $RT_N$ (voir la section 2.2.3).
- La classe *MiHarmonic* définit les éléments finis sur la variable angulaire. La variable énergétique est discrétisée en groupes. La variable angulaire des équations $SP_N$ est développée en harmoniques sur une base de polynômes de Legendre. Le flux est divisé en harmoniques paires et impaires. Pour la diffusion (correspondant aux équations $SP_0$),

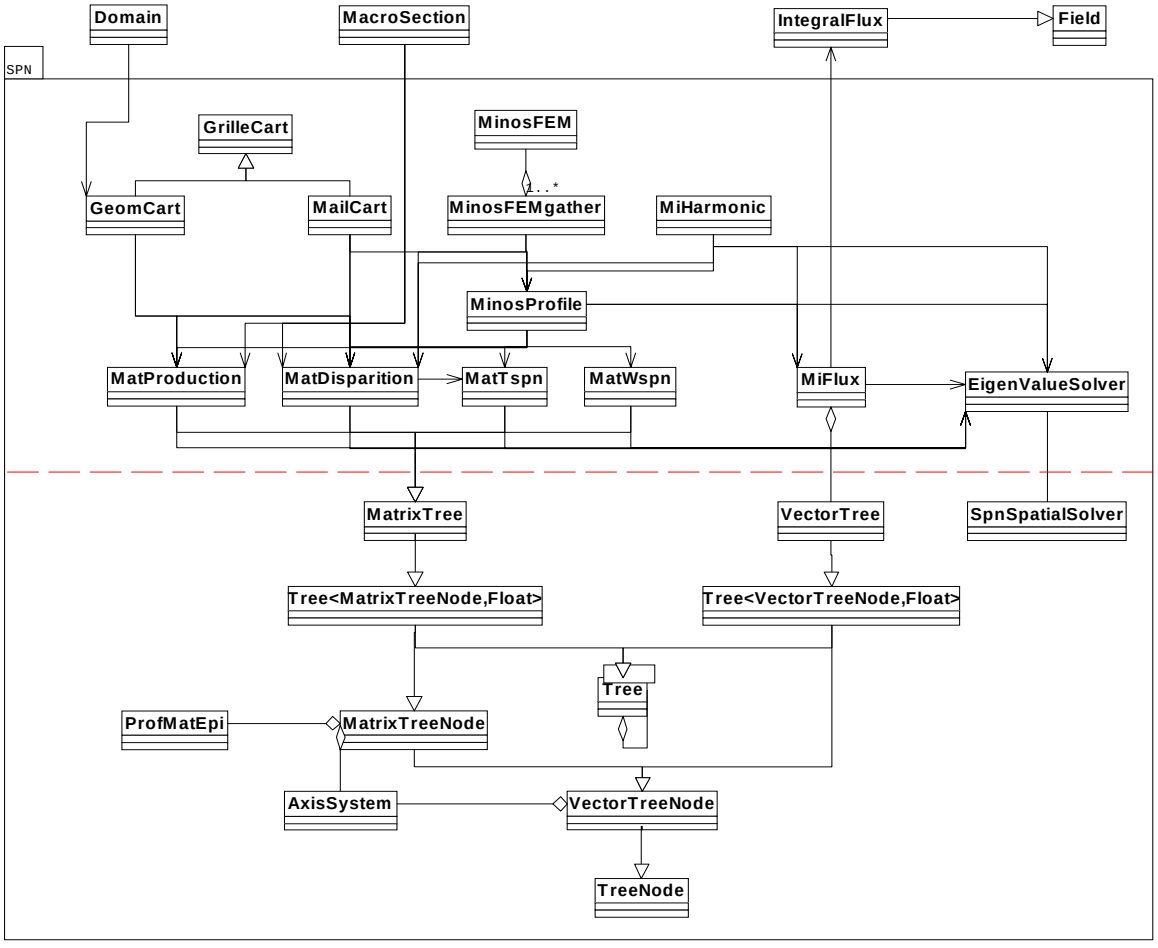



FIG. 2.4: Modèle de conception du solveur MINOS.

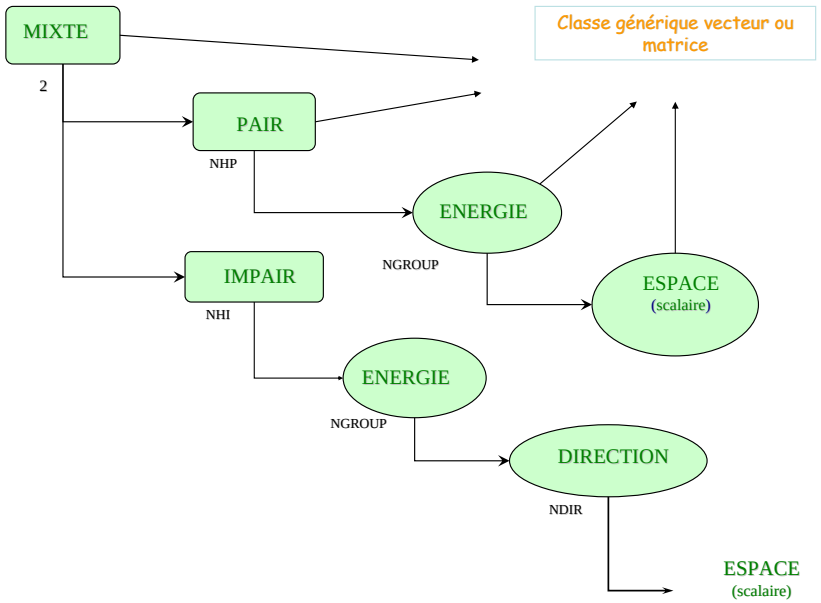



FIG. 2.5: Arborescence des classes génériques VectorTree et MatrixTree.

l'harmonique paire est le flux scalaire $\varphi$ et l'harmonique impaire est le courant $\overrightarrow{p}$.

- La classe *MinosProfile* décrit le profil des matrices à partir des informations sur les éléments finis spatiaux et angulaires données par les classes *MinosFEMGather* et *MiHarmonic*. Notamment *MinosProfile* regroupe toutes les informations nécessaires à la numérotation des nœuds de calcul et des termes des matrices.
- La classe *MiFlux* permet de stocker les flux pairs et impairs (le flux scalaire $\varphi$ et le courant $\overrightarrow{p}$ pour la diffusion).
- La classe *MatProduction* construit les matrices de production, c'est à dire les matrices liées aux termes de fission. Elles sont diagonales, seuls les termes de la diagonale sont donc stockés.
- La classe *MatDisparition* construit les matrices correspondant aux termes de transfert (dans le cas multigroupe) et aux termes de disparition, pour les flux pairs et impairs. En diffusion, la matrice de disparition sur le flux est $T$ tandis que celle sur le courant est $A_d$ (voir le système (2.40)). Ces matrices sont stockées sous forme de profil creux : seuls les termes non nuls sont stockés.
- La classe *MatTspn* utilise *MatDisparition* pour construire les matrices $T^{-1}$ qui apparaissent dans le système linéaire (2.42). Ces matrices sont associées aux sections macroscopiques de disparition paire. Elles sont diagonales en espace et calculées aux nœuds de flux pair. Seuls les termes de la diagonale sont stockés.
- La classe *MatWspn* construit les matrices $W_d$ du système (2.42). Une fonction de cette classe permet de réaliser la factorisation de Cholesky de ces matrices. Ces matrices sont symétriques, seule la partie triangulaire inférieure est stockée sous forme de profil plein, c'est à dire que tous les termes d'une ligne sont stockés à partir du premier terme non nul. La diagonale est stockée à part.
- La classe *EigenValueSolver* permet de résoudre le système linéaire (2.42) dans le cas d'un problème critique à valeur propre. L'algorithme 2 est utilisé pour déterminer la valeur propre $k_{eff}$ et le flux associés.

– En sortie de solveur la classe *IntegralFlux* fournit le flux intégré sur les mailles de *GeomCart*, et sert de paramètre à des fonctions de représentations graphiques, de calcul et d'affichage de la distribution de puissance...

### 2.3.2 Description des géométries des cœurs du REP et du RJH

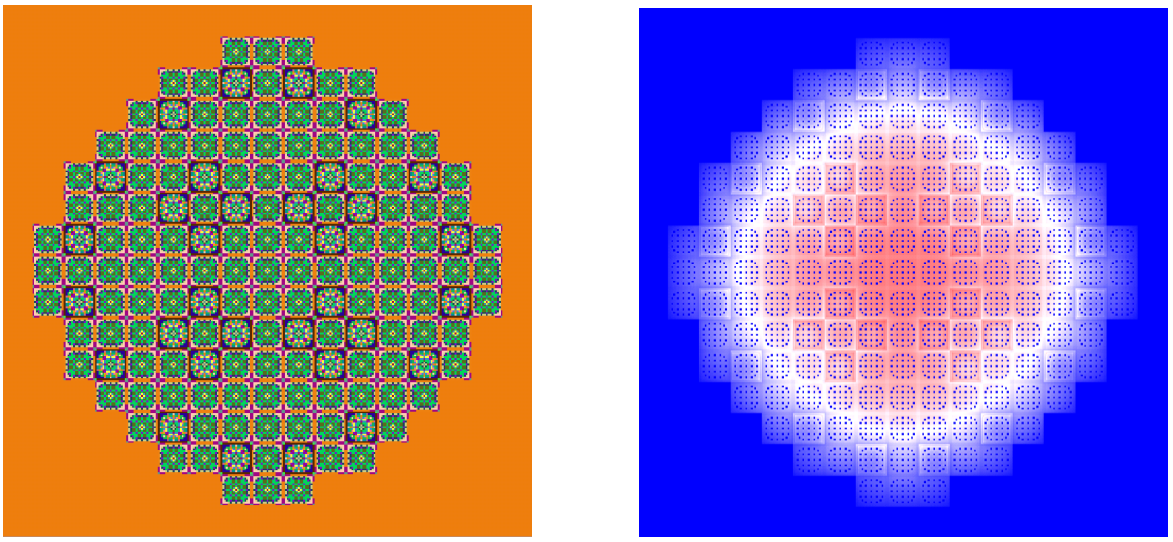
Nous présentons à présent deux géométries 2D élaborées avec DESCARTES : celle du cœur d'un Réacteur à Eau sous Pression (REP) 900 MWe, et celle du cœur du Réacteur Jules Horowitz (RJH). Ces géométries sont complexes, et permettent de valider les méthodes numériques sur des domaines de calcul réalistes et non triviaux. De plus les méthodes performantes sur de telles géométries intéressent particulièrement les industriels, car ils réalisent de nombreux calculs sur ce type de cœurs . Sur chacune des géométries, nous présentons les représentations graphiques des flux pour les différents groupes, et de la distribution de puissance. La puissance est définie par $P = \sum_{g=1}^{N_g} \nu^g \Sigma_f^g \Phi^g$. C'est une information importante pour les neutroniciens. Elle permet notamment de connaître le point chaud du cœur , qui correspond au pic de puissance. La puissance dans DESCARTES est normalisée par le volume du combustible $V_R$, telle que : $\int_{V_R} P = V_R$.

#### Le REP

Le cœur d'un Réacteur à Eau sous Pression de puissance 900 MWe a été décrit à la section 1.1.1. Il est constitué de 157 assemblages, chacun constitué de 289 crayons. Nous introduisons ici la géométrie d'un REP chargé avec deux types d'assemblage : l'assemblage UOX, qui utilise comme combustible du dioxyde d'uranium faiblement enrichi, et l'assemblage MOX fabriqué à partir d'uranium et de plutonium appauvri. La géométrie multi-échelles du cœur permet de construire le domaine en deux étapes : d'abord les assemblages UOX et MOX, ensuite le cœur décrit comme un ensemble d'assemblages.

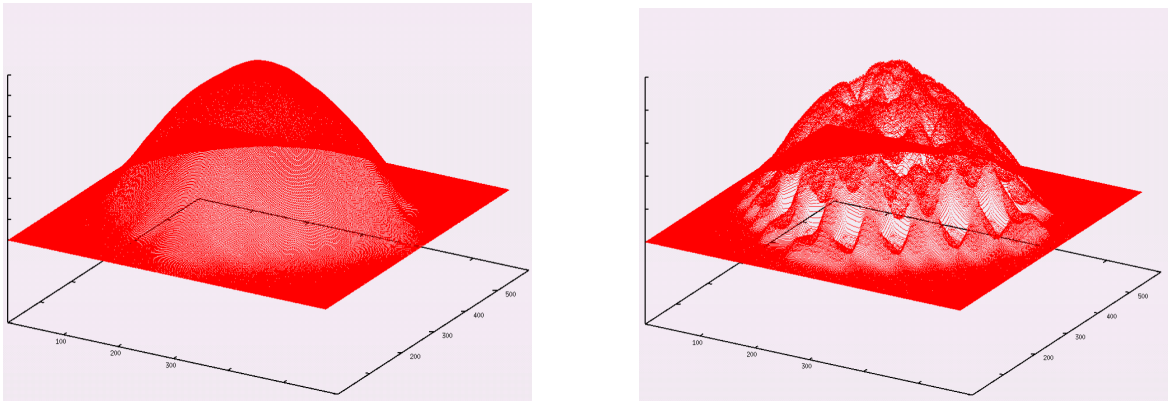
Mais la géométrie traitée par MINOS doit être cartésienne, ce qui n'est pas le cas de la géométrie réelle du REP (voir la figure 1.3). Elle est donc complétée par des assemblages fictifs de réflecteur, de manière à avoir un domaine carré constitué de 289 assemblages (17 dans chaque direction). Chaque assemblage mesure un peu moins de 22cm de côté, et le cœur complété mesure un peu moins de 3,70m de côté. La géométrie MINOS est représentée sur la figure 2.6a. Elle utilise un maillage avec la même taille de maille dans chaque direction, égale à une cellule. Une cellule est un crayon avec l'eau qui l'entoure, afin de former un ensemble carré. Ainsi un assemblage est constitué de $17^2 = 289$ cellules, et la géométrie du cœur comporte $17^4 = 83521$ mailles.

Avec le modèle de la diffusion à deux groupes d'énergie, deux flux sont calculés par le solveur MINOS : le flux rapide (figure 2.7a), correspondant aux neutrons les plus énergétiques, et le flux thermique (figure 2.7b) pour les neutrons les plus lents. Un calcul avec le même maillage que celui de la géométrie, l'élément $RT_0$ et 10000 itérations externes, aboutit à un facteur de multiplication $k_{eff} = 1.09725$. La figure 2.6b représente la distribution de puissance.



(a) Géométrie (b) Puissance

FIG. 2.6: Représentations graphiques de la géométrie et de la puissance du REP en 2D dans DESCARTES.

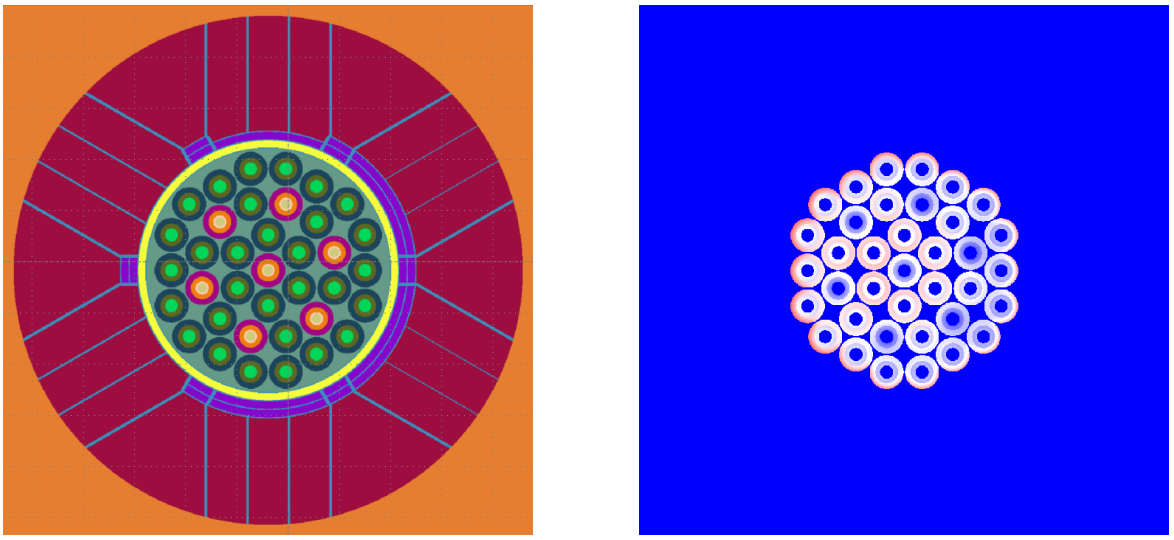


(a) Flux rapide (b) Flux thermique

FIG. 2.7: Représentations graphiques des flux du REP en 2D pour un calcul à deux groupes d'énergie.

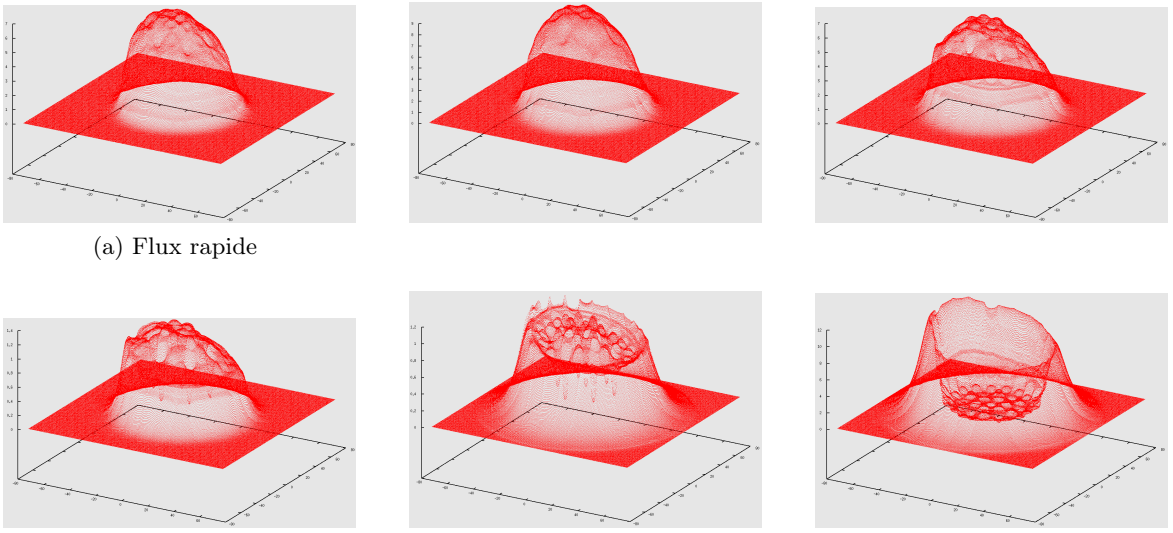
### Le RJH

Le Réacteur Jules Horowitz est un réacteur expérimental européen en cours de construction sur le site du CEA Cadarache, et qui devrait entrer en service en 2014. On remarque que, contrairement au REP, la géométrie du RJH représentée sur la figure 2.8a n'est pas cartésienne, ce qui complexifie la projection sur un maillage cartésien exigée par MINOS. Une bonne approche de la géométrie par des mailles rectangulaires demande un maillage très fin, ce qui augmente le temps de calcul et l'occupation mémoire. Le maillage géométrique utilise ici 1000 mailles par direction, le cœur totalisant donc un million de mailles. La figure 2.8b représente la distribution de puissance, tandis que la figure 2.9 représente les flux fournis par un calcul de diffusion à six groupes d'énergie. La valeur du facteur de multiplication après 50000 itérations externes est $k_{eff} = 1.30857$.



(a) Géométrie (b) Puissance

FIG. 2.8: Représentations graphiques de la géométrie et de la puissance du RJH en 2D dans DESCARTES.



(a) Flux rapide

(f) Flux thermique

FIG. 2.9: Représentations graphiques des flux du RJH en 2D pour un calcul à six groupes d'énergie.

---

*Nous venons d'exposer dans ce chapitre l'analyse numérique et les méthodes de résolution utilisées par le solveur de cœur MINOS, en limitant notre exposé à l'équation de la diffusion monocinétique, avec des conditions de flux nul aux bords du cœur . Mais MINOS est également capable de résoudre les équations $SP_N$. D'autres conditions aux bords peuvent être traitées, comme des conditions d'albedo, de courant nul, de vide, de translation, de rotation...*

*Le solveur MINOS est très rapide, et ceci s'explique notamment par l'utilisation de la formulation variationnelle mixte duale couplée à un maillage cartésien. La formulation mixte duale (2.7) permet de faire une approximation discontinue du flux, et donc d'approcher au mieux les fortes discontinuités. Elle permet également d'avoir une matrice de couplage des flux diagonale. Quant au maillage cartésien, il permet, comme nous l'avons vu à la section 2.2.8, de décomposer par direction l'algorithme de résolution. Moyennant une boucle sur les directions, les systèmes linéaires résolus par la méthode de Cholesky sont des systèmes à une dimension d'espace, avec des couplages limités aux mailles voisines. Le coût algorithmique est quasiment linéaire en fonction du nombre d'inconnues, ce qui est rarement le cas pour une méthode de type éléments finis. Le passage à des calculs 2D ou 3D n'engendre donc pas de surcoût, si ce n'est l'augmentation du nombre d'inconnues.*

*Mais la formulation mixte présente un défaut : elle introduit une inconnue auxiliaire vectorielle, le courant. En dimension 3 cela multiplie donc le nombre d'inconnues par plus de 4 (chaque composante du courant a une taille plus grande que celle du vecteur flux). Cette augmentation de la taille du système linéaire est compensée par la diminution des termes de couplage, et par le très bon conditionnement et le profil optimisé des matrices.*

*L'utilisation d'un maillage cartésien est parfois pénalisante. Dans le cas d'un réacteur de type REP, ce type de maillage est bien adapté à la géométrie cartésienne du cœur . Mais pour des géométries plus complexes et non cartésiennes, il est nécessaire d'utiliser un maillage très fin pour avoir une bonne approximation de la géométrie. De plus les sections efficaces sont homogénéisées sur chaque maille. La présence de plusieurs matériaux dans une maille peut donc s'avérer problématique. A ce sujet il est question dans [Schneider et Lautard, 2001] de l'extension du solveur MINOS aux géométries hexagonales. Nous mentionnons également les travaux récents de Jean-Jacques Lautard sur le solveur MINOS avec des éléments finis triangulaires ([Baudron et al., 2007]). L'utilisation de triangles permet de réduire significativement le nombre de mailles nécessaires pour des géométries non cartésiennes, ce qui compense le profil moins bon des matrices. La résolution du système linéaire est assurée par un gradient conjugué préconditionné (SSOR). Le nombre d'itérations internes augmente par rapport au MINOS cartésien, mais le nombre d'itérations externes diminue.*

*Un autre inconvénient de MINOS est d'être difficilement parallélisable. Nous allons voir dans le prochain chapitre que les méthodes de décomposition de domaine peuvent répondre à ce problème, et nous présenterons les techniques existantes pour paralléliser MINOS.*

# Chapitre 3

# Introduction au parallélisme et présentation de méthodes de décomposition de domaine

---

*Les calculateurs récents ont presque tous une architecture parallèle, c'est à dire basée sur plusieurs processeurs qui peuvent être exploités simultanément. Un des défis du calcul scientifique est d'adapter les codes de calcul de façon à utiliser au mieux ces calculateurs. Pour cela, des méthodes numériques facilement parallélisables peuvent être mises en œuvre, telles que les méthodes de décomposition de domaine.*

*Ce chapitre commence par une présentation du parallélisme et des machines parallèles. Les principales architectures sont décrites, ainsi que les moyens de mesurer les performances d'un code exécuté sur plusieurs processeurs. La section suivante introduit la bibliothèque MPI qui fournit un ensemble de fonctions gérant les communications entre différents processus. Puis deux méthodes de décomposition de domaine sont présentées : la méthode de synthèse modale, et l'algorithme de Schwarz. La fin du chapitre est dédiée aux différentes techniques qui ont été développées pour paralléliser MINOS.*

---

## 3.1 Notions sur le calcul parallèle

Il y a encore peu de temps, la montée en puissance des processeurs se traduisait par une augmentation de la fréquence d'horloge et par une complexification des puces et des architectures, entraînant une multiplication du nombre de transistors. Mais les fréquences et le nombre de transistors augmentant, le problème de la dissipation thermique du processeur est devenu de plus en plus critique. L'amélioration de la finesse de gravure du silicium a constitué une solution, mais ne suffit plus pour compenser la production toujours plus grande de chaleur. Depuis quelques années, un nouveau type d'architecture s'est développé, et s'impose maintenant, y compris dans les ordinateurs personnels. Le principe repose sur la multiplication des unités de traitement (cœur). Ainsi des puces à deux cœurs (figure 3.1) et à quatre cœurs ont vu le jour, et l'amélioration des performances se fait en ajoutant des cœurs plutôt qu'en augmentant la fréquence. Les prototypes de processeurs à plusieurs dizaines de cœurs semblent confirmer cette tendance, et équiperont sans doute les ordinateurs de demain.

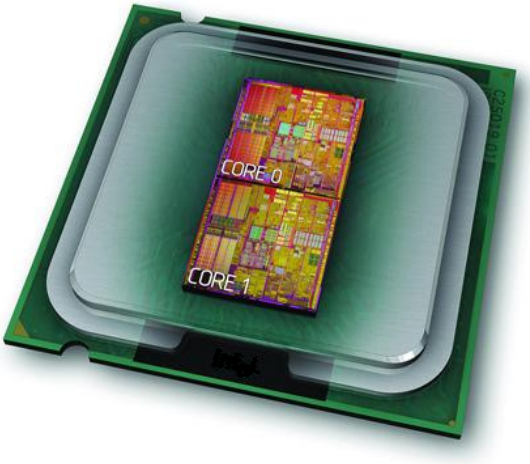



FIG. 3.1: Processeur à deux cœurs.

Dans le monde des supercalculateurs, le concept de multiplication des unités de calcul s'est imposé depuis longtemps. La dissipation de chaleur a été vite problématique, et le développement d'architectures particulièrement complexes et spécifiques a rendu les supercalculateurs très chers, réduisant le nombre de clients potentiels. L'idée d'utiliser des puces bon marché produites en masse pour les ordinateurs personnels est apparue. Plutôt que de développer des puces complexes très spécialisées, certains fabricants se sont mis à produire des supercalculateurs utilisant un grand nombre de processeurs "standards", reliés par un réseau performant pour les faire communiquer. La tendance s'est maintenant généralisée, et les machines haute performance actuelles utilisent des milliers de cœurs, comme le calculateur Tera 10 du CEA (figure 3.2), jusqu'à une centaine de milliers pour le calculateur Blue Gene américain. La question de la dissipation de chaleur se pose toujours, mais sous une autre forme. L'optimisation du rapport puissance de calcul/consommation électrique est devenue l'enjeu principal des fabricants de supercalculateurs, le développement de puces très performantes, complexes et spécialisées n'étant plus d'actualité, sauf pour des applications très spécifiques.

Mais cette évolution des architectures impose une évolution de la programmation pour tirer parti de tous les processeurs disponibles. En effet, un programme "séquentiel", prévu pour exploiter un seul processeur, n'a aucun intérêt sur une machine multi-processeurs. L'amélio-

FIG. 3.2: Le supercalculateur Tera10 du CEA.

ration des performances nécessite la parallélisation du programme : plusieurs tâches doivent être exécutées en même temps, en parallèle. Tout l'enjeu est donc d'isoler dans le programme les calculs indépendants pouvant se réaliser en même temps. Disposer de toute la puissance d'un calculateur parallèle s'avère très délicat : il faut que tous les processeurs soient sollicités constamment à 100% pendant toute la durée d'exécution d'un programme entièrement décomposable en tâches indépendantes, ce qui est très rare. Il est souvent nécessaire de prévoir des phases de communication entre ces tâches. Mais les communications entre processeurs sont relativement lentes par rapport aux capacités de calcul, il convient donc de les limiter et si possible de les "recouvrir" par des calculs.

### 3.1.1 Les modèles de programmation

Il existe principalement deux modèles de programmation parallèle : le parallélisme de tâches et le parallélisme de données. Le premier consiste à exécuter en parallèle différentes tâches du programme. Il faut que ces tâches soient relativement indépendantes pour qu'il soit possible de les exécuter en même temps sans nécessiter trop de communications entre elles. Le principe du parallélisme de données est de distribuer les données sur les processeurs et d'exécuter les mêmes tâches sur ces données. Nous aborderons essentiellement cette technique de programmation dans notre travail : les méthodes de décomposition y amènent naturellement. Là aussi, il faut veiller à ce qu'il n'y ait pas trop d'interactions entre les calculs sur les différentes données, de façon à limiter les communications.

### 3.1.2 Les architectures des ordinateurs parallèles

On distingue deux types d'architecture multi-processeurs. Le premier est dit à mémoire partagée, car les processeurs ont la même mémoire vive. L'accès à cette mémoire est rapide, et les différents processus peuvent éventuellement accéder aux mêmes adresses. Il est donc possible d'éviter les recopies de données, mais un point délicat est la gestion des conflits d'accès à la mémoire entre les processus. Ce type de calculateur est cher, et limité à un nombre restreint de processeurs.

Le deuxième type d'architecture est dit à mémoire distribuée : chaque processeur possède sa propre mémoire, et est relié aux autres via un réseau rapide. Il n'a donc accès qu'à ses

données, et pas à celles présentes dans la mémoire des autres processeurs (bien qu'il existe des environnements d'exploitation permettant de simuler une mémoire commune). Un échange de données d'un processeur à un autre se fera donc nécessairement par recopie des données de la mémoire locale de l'émetteur vers celle du récepteur, via le réseau de communication. Ce type d'architecture permet d'avoir une machine composée d'un grand nombre de processeurs, mais l'utilisation d'un réseau performant est indispensable de façon à ne pas avoir des temps de communication trop grands.

Un grand nombre de supercalculateurs utilise en fait un mélange de ces deux architectures : des nœuds de calcul composés de plusieurs processeurs en mémoire partagée sont reliés entre eux par un réseau. C'est la cas du calculateur Tantale du CCRT, que nous allons utiliser par la suite. Il est composé de 128 nœuds de quatre processeurs *AMD Opteron* cadencés à 1,8 ou 2,4 GHz avec 4 Go de mémoire partagée, et de 10 nœuds avec 32 Go de mémoire. Les nœuds sont reliés par un réseau rapide *Infiniband*. Le calculateur Lucretia du SERMA est également basé sur des nœuds de quatre processeurs *AMD Opteron* simple ou bicœurs, mais ne possède que six nœuds reliés par un réseau Gigabit, plus lent que l'Infiniband. Sa particularité est que chaque nœud possède une grande quantité de mémoire (de 32 à 128 Go chacun).

### 3.1.3 Mesure des performances

Plusieurs indicateurs permettent de mesurer les performances d'un programmme parallélisé. Entre autres, l'efficacité d'un code parallèle est la valeur sans dimension définie par :

$$Eff = \frac{T_1}{N \times T_N}, \tag{3.1}$$

où $T_1$ est le temps d'exécution du code sur un processeur, $N$ est le nombre de processeurs et $T_N$ est le temps d'exécution du code sur $N$ processeurs. Une efficacité de 1 indique que l'exécution est $N$ fois plus rapide sur $N$ processeurs que sur un seul. On peut observer dans certains cas une efficacité supérieure à 1, notamment due à une meilleure utilisation de la mémoire cache des processeurs. Ainsi on peut évaluer la "scalabilité" du code, c'est à dire sa capacité à conserver une bonne efficacité lorsque le nombre de processeurs augmente.

Une bonne efficacité est en général obtenue en limitant et en optimisant les communications. On distingue deux modes de communication :

- les communications bloquantes : l'exécution du code s'arrête à l'instruction de communication et ne reprend que lorsque les émissions et les réceptions sont complètes ;
- les communications non bloquantes : l'instruction de communication est prise en compte, mais l'exécution continue et passe aux instructions suivantes, la communication se faisant en tâche de fond. C'est alors à l'utilisateur de vérifier si la communication s'est bien déroulée.

Ainsi, une bonne gestion de la parallélisation passe souvent par l'utilisation de communications non bloquantes, que l'on peut recouvrir par des calculs n'utilisant pas les valeurs communiquées. Par exemple, on peut lancer la communication d'un vecteur dès que celui-ci est disponible, et recouvrir le temps de communication par un ou plusieurs calculs ne faisant pas appel à ce vecteur. Avant l'utilisation du vecteur par le récepteur, sa bonne réception doit être vérifiée, avec une attente bloquante si elle n'est pas terminée. De cette manière, il est possible dans certains cas de figure d'avoir un très bon recouvrement des communications. Elles n'ont alors presque pas d'impact sur le temps de calcul, ce qui permet d'avoir une efficacité voisine de 1.

La mesure du temps d'exécution d'un programme parallèle est délicate, puisqu'elle concerne un ensemble de processeurs. Dans notre travail, nous considérerons toujours le temps

réel d'exécution, c'est à dire le temps écoulé entre le début et la fin de l'exécution du programme. Un inconvénient de cette façon de procéder est que l'on prend en compte le temps où les processeurs exécutent d'autres tâches (lancées par le système ou par d'autres utilisateurs). Il convient donc de s'assurer que les processeurs utilisés sont "dédiés" au programme dont on veut connaître le temps d'exécution. Des outils permettent une analyse plus fine afin d'isoler les temps correspondant aux communications, aux calculs, aux accès disques, aux interruptions système...

L'efficacité définie par l'équation (3.1) suppose que l'algorithme exécuté en parallèle et en séquentiel est le même : les opérations réalisées doivent être identiques. Mais beaucoup de méthodes permettant de paralléliser la résolution numérique d'un problème ne sont pas équivalentes à la méthode séquentielle. C'est par exemple le cas des méthodes de décomposition de domaine, qui n'aboutissent pas toujours au même algorithme lorsque le nombre de sous-domaines est différent. L'efficacité n'est alors pas la même que celle donnée par la définition (3.1). En effet on ne mesure plus les temps d'exécution du même algorithme : $T_1$ est le temps d'exécution de l'algorithme séquentiel sur un processeur, tandis que $T_N$ désigne le temps d'exécution de l'algorithme parallèle sur N processeurs. Cette notion d'efficacité n'a un sens que pour des algorithmes qui fournissent des qualités d'approximation comparables. Nous l'utilisons dans la partie II de ce manuscrit pour illustrer les performances des méthodes de décomposition de domaine que nous avons développées.

### 3.1.4 La bibliothèque MPI

Plusieurs bibliothèques permettent de gérer les communications entre processus dans un programme parallèle. Nous utiliserons dans notre travail la bibliothèque MPI (*Message Passing Interface*), qui gère aussi bien les architectures à mémoire partagée que celles à mémoire distribuée. MPI marche sur le principe des messages postaux : un message, composé de plusieurs données de même type stockées dans la mémoire du processus expéditeur, est envoyé via le réseau de communication. Le processus récepteur le copie après sa réception dans sa propre mémoire. On distingue les communications point à point et les communications collectives. Les fonctions *MPI_Send* et *MPI_Recv* correspondent à un envoi point à point synchrone : un *MPI_Send* de l'émetteur doit correspondre à un *MPI_Recv* du récepteur de manière synchronisée et bloquante. *MPI_Isend* et *MPI_Irecv* correspondent à une communication asynchrone et non bloquante, mais le récepteur doit s'assurer de la bonne arrivée du message. Les fonctions MPI pour les communications point à point utilisent plusieurs paramètres définissant le contexte de la communication :

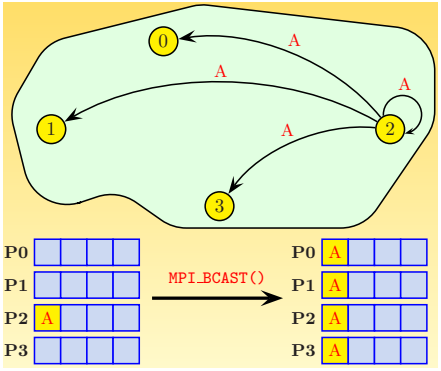
- le nom du communicateur, qui correspond à un ensemble de processus. Le communicateur *MPI_COMM_WORLD* inclut l'ensemble des processus ;
- le rang du processus émetteur, qui permet d'identifier le processus dans le communicateur ;
- le rang du processus récepteur ;
- l'étiquette du message (*tag*) qui permet d'identifier la communication.

Pour les communications collectives (figure 3.3), plusieurs fonctions MPI permettent de réaliser des échanges entre tous les processus d'un communicateur passé en paramètre :

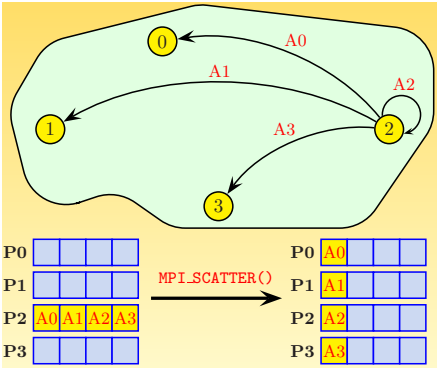
- *MPI_Bcast* permet à un émetteur d'envoyer un message à un groupe de récepteurs ;
- *MPI_Scatter* diffuse plusieurs messages de manière sélective d'un émetteur vers plusieurs récepteurs ;
- *MPI_Gather* regroupe vers un récepteur plusieurs messages venant d'émetteurs différents ;

- *MPI_ Allgather* permet de diffuser les messages de plusieurs émetteurs vers un ensemble de récepteurs ;
- *MPI_ Alltoall* est un échange croisé de plusieurs messages de plusieurs émetteurs vers plusieurs récepteurs ;
- *MPI_ Reduce* permet de faire une opération de réduction (somme, produit, min, max...) à partir de messages de plusieurs émetteurs. Le résultat est collecté par un récepteur ;
- *MPI_ Allreduce* est un *MPI_ Reduce* suivi d'un *MPI_ Bcast*, permettant ainsi de diffuser le résultat à un ensemble de récepteurs ;
- *MPI_ Barrier* assure la synchronisation de plusieurs processus.

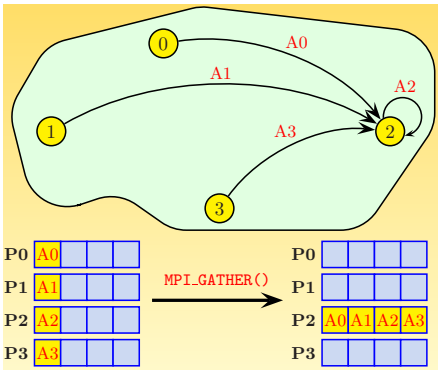
Nous signalons également la bibliothèque OpenMP, qui gère le parallélisme uniquement en mémoire partagée, mais qui est simple d'utilisation. Il est possible d'utiliser OpenMP à l'intérieur des nœuds , et de faire communiquer les nœuds avec MPI.



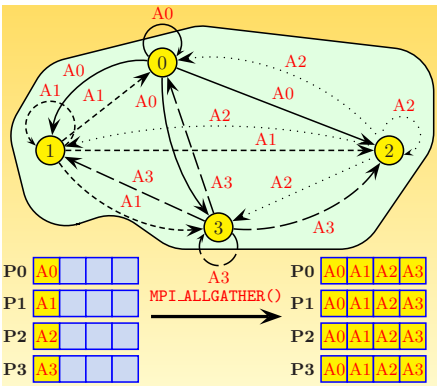

(a) *MPI_ Bcast*



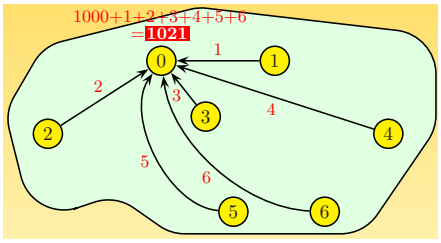

(b) *MPI_ Scatter*



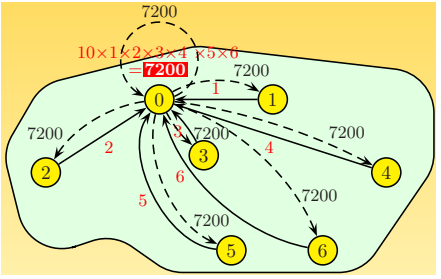

(c) *MPI_ Gather*



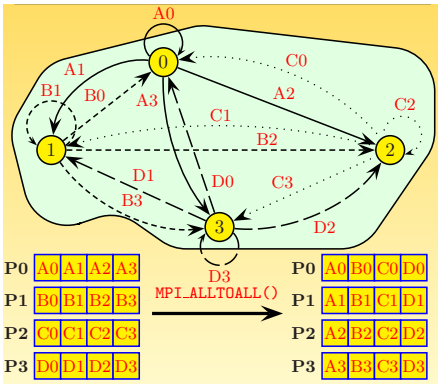

(d) *MPI_ Allgather*




(e) *MPI_ Reduce*




(f) *MPI_ Allreduce*




(g) *MPI_ Alltoall*

FIG. 3.3: Communications collectives MPI.

## 3.2 Présentation de deux méthodes de décomposition de domaine

Les méthodes de décomposition de domaine ont pour but de découper un problème défini sur un domaine $R$ en K sous-problèmes définis sur des sous-domaines $R^k$, tels que : $R = \bigcup_{k=1}^{K} R^k$. Ces K sous-domaines peuvent se recouvrir ou non, en fonction de la méthode utilisée. Sur chaque sous-domaine $R^k$, des problèmes locaux de même nature que le problème global sont résolus. Le couplage des problèmes locaux se fait alors au niveau des interfaces ou des recouvrements entre les sous-domaines. La parallélisation de ces méthodes est alors naturelle : les sous-domaines et les calculs locaux correspondants sont répartis sur différents processeurs. Le but est de limiter les surcoûts des calculs locaux par rapport au calcul global (engendrés par exemple par les recouvrements des sous-domaines), et de minimiser les couplages entre sous-domaines, qui impliquent en général des communications entre processus. Notons que certaines méthodes de décomposition de domaine ont un équivalent algébrique : les matrices du système linéaire associé au problème global sont décomposées en blocs correspondant aux sous-domaines, et en blocs correspondant aux interfaces.

Nous présentons deux méthodes de décomposition de domaine sur lesquelles sont basées nos travaux. La première est la méthode de synthèse modale, qui permet de résoudre des problèmes à valeurs propres. La deuxième est une modification de l'algorithme de Schwarz, et permet de résoudre des problèmes à source.

Nous emploierons dorénavant le terme "local" pour évoquer les problèmes sur les sous-domaines, alors que le terme "global" fera référence au problème sur le domaine complet.

### 3.2.1 La méthode de synthèse modale

La méthode de synthèse modale (appelée CMS, pour *Component Mode Synthesis method*) est une méthode de décomposition de domaine qui permet d'approcher les solutions propres de problèmes elliptiques à valeurs propres du type :

$$Au_i = \lambda_i u_i \quad \text{sur } R, \tag{3.2}$$

avec $\lambda_i$ la i-ème valeur propre du problème, $u_i$ le vecteur propre associé et $A$ un opérateur elliptique. Elle est notamment utilisée dans des problèmes de mécanique des structures, par exemple pour la recherche des modes résonnants d'une structure complexe. La technique d'approximation est de type Galerkin : le problème (3.2) est discrétisé dans un espace de dimension finie $V_\delta$, et les modes propres solutions de ce problème discret sont une projection sur $V_\delta$ de modes propres du problème (3.2). La qualité de l'approximation dépend donc essentiellement du choix de $V_\delta$.

Le principe de la méthode de synthèse modale est d'engendrer l'espace d'approximation $V_\delta$ à partir de modes propres locaux de l'opérateur $A$. Le domaine $R$ est décomposé en sous-domaines $R^k$ tels que $R = \bigcup_{k=1}^{K} R^k$, et $N^k$ modes propres $(u_i^k, \lambda_i^k)$ de l'opérateur $A$ sont calculés sur chaque sous-domaine $R^k$ :

$$Au_i^k = \lambda_i^k u_i^k \quad \text{sur } R^k, 1 \leq i \leq N^k. \tag{3.3}$$

L'idée originale utilise des sous-domaines non recouvrants. Le couplage des modes propres locaux est assuré par $N^\Gamma$ modes $(\omega_j^\Gamma)_{1 \leq j \leq N^\Gamma}$ d'interface $\Gamma$ entre les sous-domaines. L'espace

d'approximation $V_\delta$ est alors défini par :

$$V_\delta = \bigoplus_{k=1}^{K} \mathcal{B}^k \bigoplus \mathcal{B}^\Gamma, \tag{3.4}$$

avec :

$$\mathcal{B}^k = Vect\left\{\tilde{u}_i^k\right\}_{1\leq i\leq N^k}, \quad \mathcal{B}^\Gamma = Vect\left\{\omega_j^\Gamma\right\}_{1\leq j\leq N^\Gamma}, \tag{3.5}$$

$\tilde{u}_i^k$ étant les solutions de (3.3) prolongées par zéro sur $R\setminus R^k$ de manière à être définies sur $R$.

Plusieurs variantes existent pour la détermination des modes d'interface $\omega_j^\Gamma$. Dans [Craig et Bampton, 1968], des fonctions de forme (du type de celles utilisées dans la méthode des éléments finis) prises aux nœuds de l'interface sont utilisées. L'inconvénient de ce choix est que le nombre de modes d'interface est proportionnel au nombre de nœuds de l'interface. La technique proposée dans [Bourquin, 1992] repose sur l'utilisation de modes d'interface solutions propres d'un problème de Poincaré-Steklov. L'utilisation de sous-domaines non recouvrants est un avantage, mais la nécessité d'introduire des modes d'interface est pénalisante : la méthode est complexe à mettre en œuvre, et son taux de convergence est fini.

Une deuxième approche, proposée dans [Charpentier *et al.*, 1996], utilise au contraire des sous-domaines recouvrants, en supprimant les modes d'interface : l'espace d'approximation est constitué uniquement d'un nombre fini de modes propres sur chaque sous-domaine. Le couplage des modes est assuré par le recouvrement des sous-domaines. Outre la plus grande facilité d'utilisation que cela apporte, le taux de convergence infini de la méthode est montré dans [Charpentier *et al.*, 1996] pour le problème à valeur propre suivant :

$$\begin{cases} -\Delta u_i = \lambda_i u_i \quad \text{sur } R, \\ u_i = 0 \quad \text{sur } \partial R. \end{cases} \tag{3.6}$$

### 3.2.2 L'algorithme de Schwarz

#### Le problème continu

Nous nous plaçons dans le cadre du problème de Laplace à source suivant :

$$\begin{cases} -\Delta u = f \quad \text{sur } R, \\ u = 0 \quad \text{sur } \partial R. \end{cases} \tag{3.7}$$

On suppose que le domaine $R$ est décomposé en deux sous-domaines non recouvrants (figure 3.4) : $R = R^1 \cup R^2$, avec $\mathring{R}^1 \cap \mathring{R}^2 = \emptyset$ et $\Gamma = \partial R^1 \cap \partial R^2$. Si l'on pose $u^k = u_{|R^k}$, le problème (3.7) est équivalent au problème de transmission suivant (voir [Dautray et Lions, 1988]) :

$$\begin{cases} -\Delta u^k = f \quad \text{sur } R^k, \quad k = 1, 2, \\ u = 0 \quad \text{sur } \partial R^k \setminus \Gamma, \quad k = 1, 2, \\ u^1 = u^2 \quad \text{sur } \Gamma, \\ \dfrac{\partial u^1}{\partial n^1} = -\dfrac{\partial u^2}{\partial n^2} \quad \text{sur } \Gamma. \end{cases} \tag{3.8}$$

avec $n^1, n^2$ les normales sortantes aux sous-domaines $R^1$ et $R^2$ sur $\Gamma$. Les solutions locales $u^1$ et $u^2$ doivent donc avoir les mêmes valeurs et les mêmes dérivées normales sur $\Gamma$ pour être les restrictions de la solution de (3.7). L'algorithme de Schwarz et les algorithmes qui s'en inspirent sont itératifs, et ont pour principe de résoudre à chaque itération des problèmes locaux sur chaque sous-domaine avec des conditions aux interfaces déduites des solutions sur les sous-domaines voisins à l'itération précédente. Ces algorithmes convergent vers la solution globale en assurant la continuité des solutions locales et de leurs dérivées normales aux interfaces des sous-domaines.

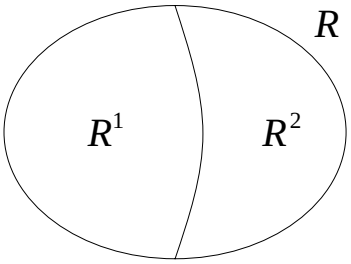



FIG. 3.4: Décomposition de domaine en deux sous-domaines non recouvrants.

**L'algorithme original**

L'algorithme de Schwarz a été développé en 1870 pour l'étude de l'opérateur de Laplace, et constitue la première méthode de décomposition de domaine. L'idée est d'étudier le problème de Laplace (3.7) sur un domaine $R$ décomposé en deux sous-domaines recouvrants : $R = R^1 \cup R^2$, avec $R^1 \cap R^2 \neq \emptyset$. Un exemple d'une telle décomposition est donné par la figure 3.5. Le principe de l'algorithme de Schwarz est de résoudre séparément un problème de Laplace sur

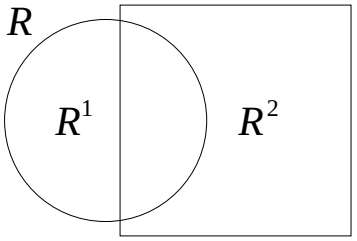



FIG. 3.5: Décomposition de domaine en deux sous-domaines recouvrants.

chaque sous-domaine, en utilisant à chaque itération (d'indice $n$) une condition de Dirichlet à l'interface déduite de la solution calculée sur le sous-domaine voisin :

$$\left\{\begin{array}{l} -\Delta u^1_{n+1} = f \quad \text{sur } R^1, \\ u^1_{n+1} = 0 \quad \text{sur } \partial R^1 \cap \partial R, \\ u^1_{n+1} = u^2_n \quad \text{sur } \partial R^1 \cap \bar{R^2}. \end{array}\right. \qquad \left\{\begin{array}{l} -\Delta u^2_{n+1} = f \quad \text{sur } R^2, \\ u^2_{n+1} = 0 \quad \text{sur } \partial R^2 \cap \partial R, \\ u^2_{n+1} = u^1_{n+1} \quad \text{sur } \partial R^2 \cap \bar{R^1}. \end{array}\right. \tag{3.9}$$

Cet algorithme, dit multiplicatif, est séquentiel, car le problème sur $R^1$ doit être résolu avant celui sur $R^2$. Au contraire, la variante de cet algorithme présentée ci-dessous est parallèle car les deux problèmes peuvent être résolus en même temps :

$$\left\{\begin{array}{l} -\Delta u^1_{n+1} = f \quad \text{sur } R^1, \\ u^1_{n+1} = 0 \quad \text{sur } \partial R^1 \cap \partial R, \\ u^1_{n+1} = u^2_n \quad \text{sur } \partial R^1 \cap \bar{R^2}. \end{array}\right. \qquad \left\{\begin{array}{l} -\Delta u^2_{n+1} = f \quad \text{sur } R^2, \\ u^2_{n+1} = 0 \quad \text{sur } \partial R^2 \cap \partial R, \\ u^2_{n+1} = u^1_n \quad \text{sur } \partial R^2 \cap \bar{R^1}. \end{array}\right. \tag{3.10}$$

La seule modification apportée à cet algorithme, dit additif, concerne la condition à l'interface pour le problème sur $R^2$ : on utilise $u^n_1$ et non $u^{n+1}_1$. A titre de comparaison, la différence entre les algorithmes multiplicatif et additif est de la même nature qu'entre les méthodes de Gauss-Seidel et de Jacobi pour la résolution de systèmes linéaires.

Deux inconvénients pénalisent l'algorithme de Schwarz : la nécessité du recouvrement des sous-domaines, et la lenteur de la convergence. Ces deux inconvénients sont d'ailleurs liés, puisque la convergence est d'autant plus lente que le recouvrement est faible.

### Des algorithmes modifiés

Afin d'améliorer l'algorithme, plusieurs types de modifications ont été apportées, notamment :

- remplacement des conditions de Dirichlet aux interfaces par des conditions mixtes de Robin, ou des conditions d'ordre 2. Dans ce cas la convergence ne nécessite pas de recouvrement ;
- optimisation des conditions d'interface pour améliorer la convergence ;
- remplacement de la méthode itérative par des méthodes de type Krylov.

Nous nous intéressons en particulier à l'utilisation de conditions de Robin aux interfaces. L'algorithme proposé dans [Lions, 1990] pour résoudre le problème (3.7) est le suivant :

$$\begin{cases} -\Delta u^1_{n+1} = f \quad \text{sur } R^1, \\ u^1_{n+1} = 0 \quad \text{sur } \partial R^1 \cap \partial R, \\ \left(\dfrac{\partial}{\partial n^1} + \alpha\right)(u^1_{n+1}) = \left(-\dfrac{\partial}{\partial n^2} + \alpha\right)(u^2_n) \quad \text{sur } \partial R^1 \cap \bar{R^2}. \\ -\Delta u^2_{n+1} = f \quad \text{sur } R^2, \\ u^2_{n+1} = 0 \quad \text{sur } \partial R^2 \cap \partial R, \\ \left(\dfrac{\partial}{\partial n^2} + \alpha\right)(u^2_{n+1}) = \left(-\dfrac{\partial}{\partial n^1} + \alpha\right)(u^1_n) \quad \text{sur } \partial R^2 \cap \bar{R^1}. \end{cases} \tag{3.11}$$

$\alpha$ est un coefficient strictement positif, $n^1$ et $n^2$ sont les normales extérieures à l'interface des sous-domaines $R^1$ et $R^2$. Notons que si $\alpha$ est nul, les conditions à l'interface deviennent des conditions de Neumann, et si $\alpha$ est infini ce sont des conditions de Dirichlet. Or l'algorithme ne converge pas dans ces deux cas. On voit donc que la convergence va dépendre de la valeur de $\alpha$.

On trouve dans [Lions, 1990] la démonstration de la convergence de l'algorithme (3.11) dans le cas d'un opérateur elliptique du second ordre de la forme : $A = -\Delta + b(x) \cdot \nabla + c(x)$. Cette démonstration est étendue dans [Després, 1993] à l'équation de Helmholtz, et une présentation générale est donnée dans [Collino *et al.*, 2000].

Nous nous intéressons maintenant au problème :

$$\begin{cases} \eta(x)u - \overrightarrow{\nabla}.(\kappa(x)\overrightarrow{\nabla}u) = f \quad \text{sur } R, \\ u = 0 \quad \text{sur } \partial R, \end{cases} \tag{3.12}$$

avec : $\kappa(x) \geq C > 0$ et $\eta(x) \geq 0$. On suppose que $R$ est décomposé en $K$ sous-domaines non recouvrants : $R = \bigcup_{k=1}^{K} R^k$, et $\mathring{R^k} \cap \mathring{R^l} = \emptyset, \forall k \neq l$. On note les interfaces $\Gamma^{kl} = \partial R^k \cap \partial R^l$. Les conditions aux interfaces sont plus générales que dans (3.11), car elles prennent en compte des dérivées tangentielles d'ordre 2 :

$$\kappa(x)\frac{\partial}{\partial n^k} + \alpha^{kl}(x) - \frac{\partial}{\partial \tau^k}\left(\beta^{kl}(x)\frac{\partial}{\partial \tau^k}\right), \tag{3.13}$$

avec $\alpha^{kl}$ et $\beta^{kl}$ définis sur $\Gamma^{kl}$. L'algorithme s'écrit :

$$\left\{\begin{array}{l} \eta(x)u^k_{n+1} - \overrightarrow{\nabla}.(\kappa(x)\overrightarrow{\nabla}u^k_{n+1}) = f \quad \text{sur } R^k, \\ u^k_{n+1} = 0 \quad \text{sur } \partial R \cap \partial R^k, \\ \kappa(x)\dfrac{\partial u^k_{n+1}}{\partial n^k} + \alpha^{kl}(x)u^k_{n+1} - \dfrac{\partial}{\partial \tau^k}\left(\beta^{kl}(x)\dfrac{\partial u^k_{n+1}}{\partial \tau^k}\right) \\ = -\kappa(x)\dfrac{\partial u^l_n}{\partial n^l} + \alpha^{kl}(x)u^l_n - \dfrac{\partial}{\partial \tau^l}\left(\beta^{kl}(x)\dfrac{\partial u^l_n}{\partial \tau^l}\right) \quad \text{sur } \Gamma^{kl}. \end{array}\right. \tag{3.14}$$

**Théorème 3.1** *On suppose que sur* $\Gamma^{kl}$ *:*
- $\alpha^{kl}(x) = \alpha^{lk}(x) \geq \alpha_0 > 0$ *;*
- $\beta^{kl}(x) = \beta^{lk}(x) \geq 0$.

*Alors l'algorithme (3.14) converge au sens :* $\lim_{n\to\infty} \left\| u^k_n - u_{|R^k} \right\|_{H^1(R^k)} = 0, \forall 1 \leq k \leq K$

On pourra trouver la démonstration dans [Nataf, 2001]. La convergence est également montrée pour les équations de Maxwell dans [Després *et al.*, 1992], et dans [Nataf et Rogier, 1995] pour l'équation de convection-diffusion. Cet algorithme converge aussi avec des sous-domaines recouvrants. La convergence dépend de plusieurs paramètres. En effet :
- le nombre d'itérations diminue quand la taille des recouvrements des sous-domaines augmente ;
- le nombre d'itérations augmente quand on augmente le nombre de sous-domaines ;
- les paramètres $\alpha^{kl}$ et $\beta^{kl}$ influent sur la vitesse de convergence.

## 3.3 Travaux précédents sur la parallélisation du solveur MINOS

Nous avons décrit le solveur MINOS au chapitre précédent. Ce solveur est séquentiel, puisqu'à chaque itération externe un système linéaire est résolu par une méthode de Gauss-Seidel par blocs, chaque bloc correspondant à une matrice factorisée par la méthode de Cholesky (voir section 2.2.8). La parallélisation de MINOS semble donc délicate à priori, et a été traitée notamment dans deux thèses. Nous signalons également la thèse en cours [Lathuiliere, ], qui applique une méthode de Schur dual à un solveur développé chez EDF, proche du solveur MINOS. Les premiers résultats sont très encourageants.

### 3.3.1 Une méthode de Jacobi par blocs

Dans la thèse [Coulomb, 1989], une méthode de Jacobi par blocs est développée pour le problème de la diffusion. Pour résoudre le système linéaire $Ax = b$, la matrice $A$ et le second membre $b$ sont décomposés en blocs correspondant à $K$ sous-domaines :

$$A = \begin{bmatrix} A^{1,1} & 0 & 0 & A^{1,K+1} \\ 0 & \ddots & 0 & \vdots \\ A^{K+1,1} & \dots & A^{K+1,K} & A^{K+1,K+1} \end{bmatrix}. \tag{3.15}$$

Les blocs d'indice de ligne ou de colonne égal à $K+1$ correspondent à l'ensemble des interfaces entre les sous-domaines. La méthode consiste à résoudre, à l'itération $n+1$ :

$$\left\{ \begin{array}{l} A^{1,1}x^1_{n+1} = b^1 - A^{1,K+1}x^{K+1}_n, \\ \vdots \\ A^{K+1,K+1}x^{K+1}_{n+1} = b^{K+1} - \sum_{k=1}^{K} A^{K+1,k}x^k_n. \end{array} \right. \tag{3.16}$$

Supposons que la ligne $k$ de (3.16) soit résolue par le processus $P^k$ Les communications à chaque itération sont :

$$\left\{ \begin{array}{l} x^k_{n+1} \text{ de } P^k \text{ à } P^{K+1}, \quad \forall k \leq K, \\ x^{K+1}_{n+1} \text{ de } P^{K+1} \text{ à } P^k, \forall k \leq K. \end{array} \right. \tag{3.17}$$

La méthode de Jacobi par blocs est présentée à la fois pour la formulation primale et pour la formulation mixte duale du problème de la diffusion. Cependant son intégration au solveur MINOS semble délicate, et son efficacité sur des ordinateurs parallèles n'a pas été testée.

### 3.3.2 Parallélisation par distribution des données

Dans la thèse [Pinchedez, 1999], deux méthodes de parallélisation de la résolution du problème de la diffusion des neutrons sont présentées. La première méthode consiste à distribuer l'ensemble des données sur tous les processeurs. Les matrices et les vecteurs sont décomposés en autant de composantes que de processeurs, et chaque processeur se charge des calculs correspondant aux sous-matrices et sous-vecteurs qui lui sont attribués.

L'avantage de l'approche par distribution des données est que l'algorithme est intégré au solveur MINOS. Son efficacité en parallèle est bonne pour un petit nombre de processeurs, mais elle se dégrade lorsque ce nombre augmente. La scalabilité de l'algorithme est notamment pénalisée par les communications.

### 3.3.3 Une méthode de synthèse modale sans recouvrement

La deuxième méthode développée dans [Pinchedez, 1999] est une méthode de synthèse modale pour le problème multigroupe de la diffusion critique sous forme primale (équation (1.9)). La technique utilisée, introduite section 3.2.1, est inspirée des travaux de [Bourquin, 1992]. Le domaine est décomposé en $K$ sous-domaines non recouvrants : $R = \bigcup_{k=1}^{K} R^k$, et $\mathring{R}^k \cap \mathring{R}^l = \emptyset, \forall k \neq l$. Sur chaque sous-domaine $R^k$, pour chaque groupe d'énergie, les $N^k$ premiers modes propres $(u^k_i, \lambda^k_i)$ du problème de diffusion suivant sont calculés :

$$\left\{ \begin{array}{ll} -\overrightarrow{\nabla}.(D\overrightarrow{\nabla}u^k_i) + \sigma_a u^k_i = \dfrac{1}{\lambda^k_i} u^k_i & \text{sur } R^k, \\ u^k_i = 0 & \text{sur } \partial R^k. \end{array} \right. \tag{3.18}$$

L'ensemble de ces modes propres prolongés par zéro sur $R \setminus R^k$, auxquels on ajoute des modes d'interface solutions propres d'un opérateur d'interface, forme l'espace d'approximation $V_\delta$ dans lequel est résolu le problème multigroupe (1.9). Le problème (3.18) utilise pour l'approximation de l'opérateur multigroupe de la diffusion, l'ensemble des opérateurs monogroupes symétriques de la diffusion. Les vecteurs propres solutions de (3.18) forment donc une base orthogonale de $H^1_0(R^k)$, ce qui n'est pas démontré pour l'opérateur multigroupe

non-autoadjoint de la diffusion. L'inconvénient de cette approche est que les équations du problème (3.18) ne sont pas couplées pour les différents groupes. Les opérateurs de fission et de transfert ont été retirés par rapport au problème non-auto-adjoint (1.9), ce qui se traduit par la perte du bilan neutronique.

Des résultats numériques sur des géométries simples et homogènes en 1D et 2D permettent de valider cette méthode, et incitent à poursuivre dans cette voie car les perspectives en parallèle sont intéressantes. Mais la méthode ne permet de résoudre que la formulation primale du problème à valeur propre de la diffusion, ce qui la rend incompatible avec le solveur MINOS. De plus, l'ajout des modes d'interface rendu nécessaire par l'absence de recouvrement des sous-domaines, complexifie sa mise en œuvre .

---

*Afin d'exploiter les nouveaux calculateurs , notamment ceux disponibles au CEA et au CCRT, il est important pour les neutroniciens du CEA d'avoir à disposition des codes performants en parallèle. Plusieurs techniques ont été proposées pour paralléliser le solveur MINOS : distribution des données, méthode de Jacobi par blocs, méthode de synthèse modale sans recouvrement. La première est efficace sur un petit nombre de processeurs, mais les performances se dégradent quand ce nombre augmente. Les deux autres sont difficiles à mettre en œuvre dans le cadre de MINOS. La méthode de synthèse modale semble être bien adaptée pour la résolution du problème de la diffusion critique, et présente l'avantage d'offrir à priori une bonne scalabilité. Mais elle doit être testée sur des géométries hétérogènes avec plusieurs groupes d'énergie pour montrer son efficacité sur des cas réalistes. De plus, l'absence de recouvrement des sous-domaines rend nécessaire l'emploi de modes d'interface qui complexifient l'implémentation de la méthode.*

*Le but de notre travail de thèse est de développer de nouvelles méthodes de décomposition de domaine pour le problème critique de la diffusion sous forme mixte duale. Ces méthodes, présentées dans la deuxième partie de ce manuscrit, doivent être efficaces en parallèle. De plus, leurs interfaces avec le solveur MINOS doivent être simples, de façon à les intégrer rapidement au projet APOLLO3/DESCARTES.*

---

Deuxième partie

# Développement de deux méthodes de décomposition de domaine pour le problème à valeur propre de la diffusion des neutrons sous forme mixte duale

# Chapitre 4

# Description d'une méthode de synthèse modale

---

*Nous avons introduit la méthode de synthèse modale (CMS) à la section 3.2.1. C'est une méthode de décomposition de domaine qui fournit une approximation d'un certain nombre de modes propres d'un opérateur elliptique. L'espace d'approximation est engendré par des modes propres locaux de l'opérateur, calculés sur chaque sous-domaine de la décomposition du domaine de calcul. Deux approches existent, selon que les sous-domaines se recouvrent ou non.*

*Nous proposons dans ce chapitre une méthode de synthèse modale avec recouvrement pour résoudre le problème à valeur propre de la diffusion sous forme mixte duale. Pour cela, nous expliquons comment nous avons adapté la méthode de synthèse modale à la résolution d'un problème mixte. Nous l'appliquons ensuite au problème de la diffusion, puis nous étudions l'existence et l'unicité de la solution du problème à source correspondant. Des tests numériques sur une géométrie 1D sont présentés, afin d'étudier le comportement de la méthode, et évaluer la qualité d'approximation en fonction des différents paramètres de la méthode.*

---

## 4.1 La méthode de synthèse modale pour un problème mixte

Nous souhaitons résoudre le problème de la diffusion critique (2.7) : c'est un problème à valeur propre dont on cherche le mode fondamental. Une formulation plus générale de ce problème s'écrit sous la forme : trouver $p \in W, \varphi \in V$ et $\lambda \in \mathbb{R}$ solution fondamentale de :

$$\begin{cases} a(p,q) + b(q,\varphi) = 0 \quad \forall q \in W, \\ b(p,\psi) - t(\varphi,\psi) = \dfrac{1}{\lambda} f(\varphi,\psi) \quad \forall \psi \in V, \end{cases} \tag{4.1}$$

avec $a, b, t, f$ des formes bilinéaires continues.

On suppose que le domaine de calcul $R$ est décomposé en sous-domaines qui se recouvrent : $R = \bigcup_{k=1}^{K}$. Comme nous l'avons vu à la section 3.2.1, la méthode de synthèse modale se décompose en trois étapes : la première est le calcul des modes propres sur chaque sous-domaine, la deuxième est la détermination des matrices et du second membre du système linéaire, et la dernière est la résolution de ce système.

Afin de constituer les espaces d'approximation $W_\delta$ pour approcher $p$, et $V_\delta$ pour approcher $\varphi$, il faut calculer plusieurs modes locaux sur chaque sous-domaine. Supposons que l'on souhaite utiliser $N_\varphi^k$ modes dans $V_\delta$ et $N_p^k$ modes dans $W_\delta$. On pose $N^k = \max(N_\varphi^k, N_p^k)$. Sur chaque sous-domaine $R^k$, on détermine alors les $N^k$ premiers modes $(\varphi_i^k, p_i^k, \lambda_i^k)$ solutions de :

$$\begin{cases} a^k(p_i^k,q) + b^k(q,\varphi_i^k) = 0 \quad \forall q \in W^k, \\ b^k(p_i^k,\psi) - t^k(\varphi_i^k,\psi) = \dfrac{1}{\lambda_i^k} f^k(\varphi_i^k,\psi) \quad \forall \psi \in V^k, \end{cases} \qquad 1 \le i \le N^k. \tag{4.2}$$

Les espaces $V^k, W^k$ et les formes $a^k, b^k, t^k, f^k$ sont des restrictions de $V, W$ et $a, b, t, f$ respectivement, que nous ne préciserons pas pour rester dans un cadre général. On remarque que le nombre de modes peut différer pour chaque sous-domaine, ce qui peut être utile pour avoir une approximation plus fine sur certains domaines que sur d'autres.

Ces solutions sont définies localement sur les sous-domaines. On les prolonge par zéro de façon à avoir des fonctions $\tilde{p}_i^k, \tilde{\varphi}_i^k$ définies sur le domaine R :

$$\begin{cases} \tilde{p}_i^k = p_i^k \quad \text{sur } R^k \\ \tilde{p}_i^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \quad \begin{cases} \tilde{\varphi}_i^k = \varphi_i^k \quad \text{sur } R^k \\ \tilde{\varphi}_i^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \qquad 1 \le i \le N^k. \tag{4.3}$$

On peut dorénavant définir nos espaces d'approximation $V_\delta$ et $W_\delta$ :

$$\begin{aligned} V_\delta &= Vect \left\{ \tilde{\varphi}_i^k \right\}_{1 \le i \le N_\varphi^k}^{1 \le k \le K}, \\ W_\delta &= Vect \left\{ \tilde{p}_i^k \right\}_{1 \le i \le N_p^k}^{1 \le k \le K}. \end{aligned} \tag{4.4}$$

La conformité de la méthode impose de bien choisir les conditions aux bords des problèmes locaux (4.2), de façon à avoir $W_\delta \subset W$ et $V_\delta \subset V$. D'autre part, on a la possibilité d'avoir $N_p^k \neq N_\varphi^k$. Or le meilleur choix semble au contraire d'avoir l'égalité, de façon à ne pas calculer des solutions locales inutilisées par la suite. Mais le fait d'avoir $N_p^k > N_\varphi^k$ peut s'avérer nécessaire pour avoir une bonne approximation, comme nous l'évoquerons à la section 4.2.1.

Le problème global (4.1) discrétisé sur les espaces $V_\delta$ et $W_\delta$ est alors de trouver $p_\delta \in W_\delta$, $\varphi_\delta \in V_\delta$ et $\lambda_\delta \in \mathbb{R}$ solution fondamentale de :

$$\begin{cases} a(p_\delta,q) + b(q,\varphi_\delta) = 0 \quad \forall q \in W_\delta, \\ b(p_\delta,\psi) - t(\varphi_\delta,\psi) = \dfrac{1}{\lambda_\delta} f(\varphi_\delta,\psi) \quad \forall \psi \in V_\delta. \end{cases} \tag{4.5}$$

Comme $p_\delta \in W_\delta$ et $\varphi_\delta \in V_\delta$, on peut les écrire comme des combinaisons linéaires des fonctions de base de ces espaces :

$$\begin{aligned} p_\delta &= \sum_{k=1}^{K} \sum_{i=1}^{N_p^k} \alpha_i^k p_i^k, \\ \varphi_\delta &= \sum_{k=1}^{K} \sum_{i=1}^{N_\varphi^k} \beta_i^k \varphi_i^k. \end{aligned} \tag{4.6}$$

La deuxième étape de la méthode CMS consiste à mettre ce problème sous la forme du système linéaire suivant :

$$\begin{cases} -AP + B\Phi = 0, \\ B^T P + T\Phi = \dfrac{1}{\lambda_\delta} F\Phi. \end{cases} \tag{4.7}$$

$P$ et $\Phi$ sont les inconnues du système correspondant aux coefficients des combinaisons linéaires dans (4.6) : $P = \left[\alpha_i^k\right]_{1\le i\le N_p^k}^{1\le k\le K}$ et $\Phi = \left[\beta_i^k\right]_{1\le i\le N_\varphi^k}^{1\le k\le K}$.

Les termes des matrices $A, B, T, F$ correspondent à l'application des formes $a, b, t, f$ respectivement, sur les fonctions de base de $V_\delta$ et $W_\delta$. Ces fonctions de base étant les modes propres solutions des problèmes locaux (4.2) prolongés par zéro, les matrices sont divisées en blocs. Chaque bloc est lié à un sous-domaine, ou à l'interface entre deux sous-domaines puisque les sous-domaines se recouvrent. Les termes de chaque bloc correspondent aux différents modes calculés sur le ou les deux sous-domaines associés au bloc. Ainsi, les matrices $A, B, T, F$ ont la forme suivante :

$$M = \begin{pmatrix} M^{11} & M^{12} & \cdots & M^{1K} \\ M^{21} & M^{22} & \cdots & M^{2K} \\ \vdots & \vdots & \ddots & \vdots \\ M^{K1} & M^{K2} & \cdots & M^{KK} \end{pmatrix}. \tag{4.8}$$

Les blocs des différentes matrices sont donnés par :

$$\begin{aligned} \left(A^{kl}\right)_{i,j} &= a(p_i^k, p_j^l); \\ \left(B^{kl}\right)_{i,j} &= b(p_i^k, \varphi_j^l); \\ \left(T^{kl}\right)_{i,j} &= t(\varphi_i^k, \varphi_j^l); \\ \left(F^{kl}\right)_{i,j} &= f(\varphi_i^k, \varphi_j^l). \end{aligned} \tag{4.9}$$

La dernière étape de la méthode de synthèse modale est la résolution du système linéaire (4.7).

## 4.2 Application au problème de la diffusion critique mixte duale

Nous allons maintenant présenter l'application de la méthode de synthèse modale pour la résolution du problème à valeur propre de la diffusion mixte duale (2.7). Le lien entre ce problème et le problème abstrait (4.1) est donné par l'expression des formes $a, b, t$ dans (2.10). Quant à la forme $f(\varphi, \psi)$, elle correspond à $\displaystyle\int_R \sigma_f \varphi\psi$.

Une fois la décomposition de domaine choisie, la première étape est de résoudre des problèmes locaux de diffusion sur chaque sous-domaine, avec des conditions aux bords à préciser. Ces conditions doivent permettre d'avoir une approximation conforme : la prolongation par zéro des solutions locales de courant doit être telle que $W_\delta \subset H(div, R)$. De même, on souhaite

que l'espace $V_\delta$ engendré par les solutions locales de flux prolongées par zéro soit tel que : $V_\delta \subset L^2(R)$. Cette dernière condition est naturellement vérifiée, quelques soient les traces des fonctions locales de flux aux bords des sous-domaines. Par contre l'inclusion de $W_\delta$ dans $H(div, R)$ implique que la trace normale du courant soit continue aux interfaces. On impose donc sur tous les bords "internes" $\partial R^k \setminus \partial R$, des conditions de courant nul. On garde par contre sur les bords du cœur $\partial R$ les conditions définies par le problème global, en l'occurrence des conditions de flux nul.

Les problèmes locaux à résoudre sur tous les sous-domaines $R^k$ sont donc de trouver les $N^k$ premiers modes propres $p_i^k \in H_{0,\partial R^k \setminus \partial R}(div, R^k), \varphi_i^k \in L^2(R^k)$ et $\lambda_i^k \in \mathbb{R}$ solutions de :

$$\left\{ \begin{array}{ll} \displaystyle\int_{R^k} -\frac{1}{D}\overrightarrow{p_i^k} \cdot \overrightarrow{q} + \int_{R^k} \overrightarrow{\nabla}.\overrightarrow{q}\,\varphi_i^k = 0 & \forall \overrightarrow{q} \in H_{0,\partial R^k \setminus \partial R}(div, R^k), \\ \displaystyle\int_{R^k} \overrightarrow{\nabla}.\overrightarrow{p_i^k}\psi + \int_{R^k} \sigma_a \varphi_i^k \psi = \frac{1}{\lambda_i^k}\int_{R^k} \sigma_f \varphi_i^k \psi & \forall \psi \in L^2(R^k), \end{array} \right. \tag{4.10}$$

avec $1 \leq k \leq K, 1 \leq i \leq N^K$. L'espace $H_{0,\partial R^k \setminus \partial R}(div, R^k)$ a été introduit dans la section 2.1.1, et correspond aux fonctions de $H(div, R)$ dont la trace normale s'annule sur $\partial R^k \setminus \partial R$.

On dénote par un "˜" le prolongement par zéro des solutions locales de flux et de courant. Afin d'avoir des fonctions de base de courant avec une seule composante non nulle correspondant à une direction donnée, on décompose les solutions locales en autant de fonctions qu'il y a de directions. Par exemple en 3D, on pose :

$$\overrightarrow{\tilde{p}_{i,x}^k} = \begin{bmatrix} \tilde{p}_{i,x}^k \\ 0 \\ 0 \end{bmatrix}, \quad \overrightarrow{\tilde{p}_{i,y}^k} = \begin{bmatrix} 0 \\ \tilde{p}_{i,y}^k \\ 0 \end{bmatrix}, \quad \overrightarrow{\tilde{p}_{i,z}^k} = \begin{bmatrix} 0 \\ 0 \\ \tilde{p}_{i,z}^k \end{bmatrix} \quad \text{avec } \tilde{p}_{i,d}^k = \overrightarrow{\tilde{p}_i^k} \cdot \overrightarrow{d}\,. \tag{4.11}$$

Les espaces discrets de flux et de courant sont alors définis par :

$$\begin{array}{l} W_\delta = Vect\left\{\overrightarrow{\tilde{p}_{i,d}^k}\right\}_{1\leq i\leq N_p^k, 1\leq d\leq N_d}^{1\leq k\leq K} \subset H(div, R), \\ V_\delta = Vect\left\{\tilde{\varphi}_i^k\right\}_{1\leq i\leq N_\varphi^k}^{1\leq k\leq K} \subset L^2(R). \end{array} \tag{4.12}$$

Comme nous l'avons vu, la conformité est garantie par les conditions de courant nul sur les bords internes des sous-domaines. Le problème discret global posé sur ces espaces est alors de trouver $p_\delta \in W_\delta$, $\varphi_\delta \in V_\delta$ et $\lambda_\delta \in \mathbb{R}$ solution fondamentale de :

$$\left\{ \begin{array}{l} \displaystyle\int_R -\frac{1}{D}\overrightarrow{p_\delta} \cdot \overrightarrow{q} + \int_R \overrightarrow{\nabla}.\overrightarrow{q}\,\varphi_\delta = 0 \quad \forall \overrightarrow{q} \in W_\delta, \\ \displaystyle\int_R \overrightarrow{\nabla}.\overrightarrow{p_\delta}\psi + \int_R \sigma_a \varphi_\delta \psi = \frac{1}{\lambda_\delta}\int_R \sigma_f \varphi_\delta \psi \quad \forall \psi \in V_\delta. \end{array} \right. \tag{4.13}$$

Les inconnues de flux et de courant de ce problème peuvent s'écrire comme une combinaison linéaire des fonctions de base :

$$\overrightarrow{p_\delta} = \sum_{k=1}^{K}\sum_{i=1}^{N^k}\sum_{d=1}^{N_d} c_{i,d}^k \overrightarrow{\tilde{p}_{i,d}^k}, \quad \varphi_\delta = \sum_{k=1}^{K}\sum_{i=1}^{N^k} f_i^k \tilde{\varphi}_i^k. \tag{4.14}$$

Le problème (4.13) est équivalent au système linéaire suivant :

$$\left\{ \begin{array}{l} -A_d\left[c_{i,d}^k\right] + B_d\left[f_i^k\right] = 0, \quad 1 \leq d \leq N_d, \\ \displaystyle\sum_{d=1}^{N_d} B_d^T\left[c_{i,d}^k\right] + T_a\left[f_i^k\right] = \frac{1}{\lambda_\delta} T_f\left[f_i^k\right]. \end{array} \right. \tag{4.15}$$

En l'écrivant comme un unique système linéaire avec pour inconnues $P = \left[c_{i,d}^k\right]_{1\leq i\leq N_p^k, 1\leq d\leq N_d}^{1\leq k\leq K}$ et $\Phi = \left[f_i^k\right]_{1\leq i\leq N_\varphi^k}^{1\leq k\leq K}$, on obtient un système linéaire équivalent au système (2.40) résolu par MINOS. Sa dimension est donnée par celles des espaces : $\sum_{d=1}^{N_d}\sum_{k=1}^{K} N_p^k$ pour le courant et $\sum_{k=1}^{K} N_\varphi^k$ pour le flux. Le profil par blocs des matrices est donné par (4.8), et les termes de chaque bloc sont :

$$\begin{aligned}
\left(A_d^{kl}\right)_{i,j} &= \int_{\mathring{R}^k\cap\mathring{R}^l} \frac{1}{D}\overrightarrow{\tilde{p}_{i,d}^k}\cdot\overrightarrow{\tilde{p}_{j,d}^l};\\
\left(B_d^{kl}\right)_{i,j} &= \int_{\mathring{R}^k\cap\mathring{R}^l} \frac{1}{D}\overrightarrow{\nabla}\cdot\overrightarrow{\tilde{p}_{i,d}^k}\tilde{\varphi}_j^l;\\
\left(T_a^{kl}\right)_{i,j} &= \int_{\mathring{R}^k\cap\mathring{R}^l} \sigma_a\tilde{\varphi}_i^k\tilde{\varphi}_j^l;\\
\left(T_f^{kl}\right)_{i,j} &= \int_{\mathring{R}^k\cap\mathring{R}^l} \sigma_f\tilde{\varphi}_i^k\tilde{\varphi}_j^l.
\end{aligned} \tag{4.16}$$

Ces matrices ont un profil qui dépend du recouvrement des sous-domaines, puisque c'est ce recouvrement qui détermine les couplages. En effet les intégrales sont nulles si $\mathring{R}^k \cap \mathring{R}^l = \varnothing$. Moins les sous-domaines présentent de recouvrement, plus le profil des matrices est creux.

On remarque que notre approche diffère de celle présentée dans [Pinchedez, 1999] sur trois points. Le premier est que nous résolvons le problème sous forme mixte duale, et non sous sa forme primale : un espace d'approximation pour le courant est introduit. Le deuxième point est que nous utilisons une décomposition de domaine avec des sous-domaines recouvrants : il n'est donc pas nécessaire d'introduire des modes d'interface dans les espaces $V_\delta$ et $W_\delta$, ce qui simplifie la mise en œuvre de la méthode. Le troisième point est que les calculs locaux prennent en compte l'opérateur de fission dans le membre de droite de la deuxième équation du problème (4.10). La section efficace de fission $\sigma_f$ étant fortement discontinue, elle intervient dans la forme locale et globale de la solution. Il nous a donc paru important de la prendre en compte dans la détermination des modes locaux.

### 4.2.1 Existence, unicité du problème et qualité de l'approximation

Un problème délicat vient du fait que l'on ne peut pas appliquer directement le théorème 2.2 pour montrer l'existence et l'unicité de la solution du problème (4.13). D'une part, la forme $a$ n'est pas $KerB_h$-elliptique dans le cas discret. En effet, $KerB_h = \{q \in W_\delta, b(q,v) = 0, \forall v \in V_\delta\}$ n'est pas inclus dans son équivalent continu. Notamment, les hypothèses de la proposition 2.1 ne sont pas vérifiées : $KerB_h$ ne correspond pas aux fonctions à divergence nulle. D'autre part, la condition inf-sup (2.13) (le inf porte sur $V_\delta$ et le sup sur $W_\delta$) n'est à priori pas vérifiée. Par contre, en utilisant l'équivalence des normes $\|.\|_{L^2(R)}$ et $\|.\|_{H(div,R)}$ en dimension finie, l'ellipticité de $a$ sur $W_\delta$ est démontrée et les hypothèses du théorème 2.1 sont vérifiées.

D'autre part, il faut que $KerB_h$ ait une dimension non nulle pour que l'approximation soit correcte (voir [Roberts et Thomas, 1991] et [Brezzi et Fortin, 1991]). Une manière de le garantir est de prendre plus de modes de courant que de flux : dans ce cas $dim(W_\delta) > dim(V_\delta)$ et le noyau n'est pas réduit à zéro. Nous allons voir dans les applications numériques suivantes l'importance de ce point.

## 4.3 Estimation de l'erreur d'approximation pour la formulation primale du problème à valeur propre de la diffusion

Nous souhaitons donner une estimation de l'erreur d'approximation pour l'application de la méthode de synthèse modale au problème à valeur propre de la diffusion des neutrons. La démonstration est inspirée de celle donnée dans [Charpentier *et al.*, 1995].

### 4.3.1 Définitions

#### Présentation du problème sous forme primale

Nous nous plaçons dans le cadre du problème monocinétique de la diffusion critique sous forme primale avec des conditions de Robin aux bords :

$$\begin{cases} -\overrightarrow{\nabla}.(D\overrightarrow{\nabla}\varphi_i) + \sigma_a\varphi_i = \lambda_i\sigma_f\varphi_i \quad \text{sur } R, \\ D\dfrac{\partial\varphi}{\partial n} + \alpha\varphi_i = 0 \quad \text{sur } \Gamma, \quad \text{avec } \alpha \geq 0, \end{cases} \tag{4.17}$$

avec $\Gamma = \partial R$. Bien qu'en neutronique, seul le mode fondamental nous intéresse, on considère ici l'ensemble des solutions propres $\{\varphi_i\}_{i\geq 1}$. La formulation variationnelle de ce problème est : trouver $\varphi_i \in H^1(R)$ et $\lambda_i \in \mathbb{R}$ solutions de

$$\int_R D\overrightarrow{\nabla}\varphi_i \cdot \overrightarrow{\nabla}\psi + \alpha\int_\Gamma \varphi_i\psi + \int_R \sigma_a\varphi_i\psi = \lambda_i\int_R \sigma_f\varphi_i\psi \quad \forall\psi \in H^1(R). \tag{4.18}$$

Les hypothèses suivantes sont faites sur les sections efficaces :
- $D \in L^\infty(R), 0 < D^0 \leq D \leq D^{max} < \infty$ ;
- $\sigma_a \in L^\infty(R), 0 \leq \sigma_a \leq \sigma_a^{max} < \infty$ ;
- $\sigma_f \in L^\infty(R), 0 < \sigma_f^0 \leq \sigma_f \leq \sigma_f^{max} < \infty$.

Soit l'opérateur $A_{R,\Gamma} : H^1(R) \to L^2(R)$, défini par

$$(A_{R,\Gamma}\varphi, \psi)_{L^2(R)} = \int_R D\overrightarrow{\nabla}\varphi \cdot \overrightarrow{\nabla}\psi + \alpha\int_\Gamma \varphi\psi + \int_R \sigma_a\varphi\psi, \quad \forall\psi \in H^1(R). \tag{4.19}$$

L'opérateur $A_{R,\Gamma}$ est autoadjoint. Par ailleurs l'hypothèse d'une section efficace de fission strictement positive assure que le second membre de (4.18) est une forme bilinéaire symétrique définie positive. Cette hypothèse est forte, et en pratique jamais vérifiée puisque des matériaux non fissiles et du réflecteur sont présents dans un cœur de réacteur. Ces hypothèses nous permettent de définir les produits scalaires suivants :

$$\begin{aligned} (\varphi,\psi)_{A_{R,\Gamma}} &= (A_{R,\Gamma}\varphi,\psi)_{L^2(R)}, \quad \varphi,\psi \in H^1(R), \\ (\varphi,\psi)_{f_R} &= \int_R \sigma_f\varphi\psi, \quad \varphi,\psi \in L^2(R), \end{aligned} \tag{4.20}$$

et les normes associées :

$$\begin{aligned} \|\varphi\|^2_{A_{R,\Gamma}} &= (\varphi,\varphi)_{A_{R,\Gamma}}, \quad \varphi \in H^1(R), \\ \|\varphi\|^2_{f_R} &= (\varphi,\varphi)_{f_R}, \quad \varphi \in L^2(R). \end{aligned} \tag{4.21}$$

**La décomposition de domaine, la partition de l'unité et les problèmes locaux**

On suppose que $R$ est décomposé en K sous-domaines recouvrants : $R = \bigcup_{k=1}^{K} R^k$, et que cette décomposition de domaine est large, au sens où la distance de Hausdorff entre $\bigcup_{l\neq k} R^l \setminus R^k$ et $R^k \setminus \left(\bigcup_{l\neq k} R^l\right)$ est strictement positive. On fait l'hypothèse que la frontière de $R$ est de classe $C^\infty$. Il est alors classique de construire une partition de l'unité régulière associée à cette décomposition de domaine, c'est à dire un ensemble de fonctions $\{\chi^k\}_{1\leq k\leq K}$ positives dans $C^\infty(R)$ à support dans $R^k$ et telles que $\sum_{k=1}^{K} \chi^k \equiv 1$ sur $R$. L'hypothèse d'une décomposition de domaine large permet également d'avoir :

$$\frac{\partial^r \chi^k}{\partial n^r} = 0 \quad \text{sur } \partial R^k, \quad \frac{\partial^r \chi^k}{\partial \tau^r} = 0 \quad \text{sur } \partial R^k \setminus \partial R, \quad \forall r \geq 1, \tag{4.22}$$

avec $n$ et $\tau$ les vecteurs normal et tangent à $\partial R^k$. On appelle $\chi_R$ l'ensemble de ces partitions de l'unité. Dans le cas de domaines avec coins (leur frontière n'est pas de classe $C^\infty$), il est nécessaire d'introduire dans la partition de l'unité des fonctions "camembert" qui englobent les coins. Ces fonctions sont décrites dans [Charpentier *et al.*, 1995].

On définit également les problèmes locaux sur les sous-domaines $R^k$ :

$$\begin{cases} -\overrightarrow{\nabla}.(D\overrightarrow{\nabla}\varphi_i^k) + \sigma_a \varphi_i^k = \lambda_i^k \sigma_f \varphi_i^k \quad \text{sur } R^k, \\ \varphi_i^k = 0 \quad \text{sur } \Gamma^k, \\ D\dfrac{\partial \varphi_i^k}{\partial n} + \alpha \varphi_i^k = 0 \quad \text{sur } \Gamma_*^k, \end{cases} \tag{4.23}$$

avec $\Gamma_*^k = \partial R^k \cap \partial R$ et $\Gamma^k = \partial R^k \setminus \partial R$. La formulation variationnelle de ce problème est : trouver $\varphi_i^k \in H^1_{0,\Gamma^k}(R^k)$ et $\lambda_i^k \in \mathbb{R}$ solutions de

$$\int_{R^k} D\overrightarrow{\nabla}\varphi_i^k \cdot \overrightarrow{\nabla}\psi + \alpha \int_{\Gamma_*^k} \varphi_i \psi + \int_{R^k} \sigma_a \varphi_i^k \psi = \lambda_i^k \int_{R^k} \sigma_f \varphi_i^k \psi \quad \forall \psi \in H^1_{0,\Gamma^k}(R^k). \tag{4.24}$$

### 4.3.2 Énoncés et démonstrations de quelques lemmes

**Lemme 4.1** *Soit l'opérateur* $\dfrac{A_{R,\Gamma}}{\sigma_f} : H^1(R) \to L^2(R)$ *définit par :*

$$\left(\frac{A_{R,\Gamma}}{\sigma_f}\varphi, \psi\right)_{f_R} = (\varphi, \psi)_{A_{R,\Gamma}}, \quad \forall \varphi, \psi \in H^1(R). \tag{4.25}$$

*Soit* $\varphi_i$ *solution du problème (4.18). Alors* $\left(\dfrac{A_{R,\Gamma}}{\sigma_f}\right)^p (\varphi_i) \in H^1(R), \forall i \geq 1, \forall p \geq 0$

**Preuve** On utilise récursivement le fait que $\varphi_i \in H^1(R)$ est solution du problème aux valeurs propres suivant :

$$\left(\frac{A_{R,\Gamma}}{\sigma_f}\varphi_i, \psi\right)_{f_R} = \lambda_i (\varphi_i, \psi)_{f_R}, \quad \forall \psi \in H^1(R). \tag{4.26}$$

On a donc $\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)\varphi_i = \lambda_i\varphi_i$, et $\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)\varphi_i \in H^1(R)$.

Supposons que $\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)^n \varphi_i = \lambda_i^n\varphi_i$, ce qui implique que $\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)^n \varphi_i \in H^1(R)$. Soit $\psi \in H^1(R)$. Alors :

$$\begin{aligned}\left(\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)^{n+1}\varphi_i,\psi\right)_{f_R} &= \left(\frac{A_{R,\Gamma}}{\sigma_f}\left(\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)^n\varphi_i\right),\psi\right)_{f_R}\\ &= \lambda_i^n\left(\frac{A_{R,\Gamma}}{\sigma_f}\varphi_i,\psi\right)_{f_R}\\ &= \lambda_i^{n+1}\left(\varphi_i,\psi\right)_{f_R}.\end{aligned}$$

On a donc, $\forall p \geq 0$, $\left(\frac{A_{R,\Gamma}}{\sigma_f}\right)^p \varphi_i = \lambda_i^p\varphi_i$, et appartient donc à $H^1(R)$.

**Lemme 4.2** *La famille des solutions propres $\{\varphi_i^k\}_{i\geq 1}$ du problème (4.24) est complète dans $H^1_{0,\Gamma^k}(R^k)$. De plus, on peut normaliser les $\{\varphi_i^k\}_{i\geq 1}$ de façon à avoir $(\varphi_i^k,\varphi_j^k)_{f_{R^k}} = \delta_{ij}$. On a alors, $\forall\psi \in H^1_{0,\Gamma^k}(R^k)$,*

$$\psi = \sum_{i=1}^{\infty}\alpha_i^k\varphi_i^k, \quad \textit{avec } \alpha_i^k = \left(\psi,\varphi_i^k\right)_{f_{R^k}}. \tag{4.27}$$

On renvoie à [Babuska et Osborn, 1991] pour la démonstration. Elle est essentiellement basée sur les propriétés des formes bilinéaires décrites à l'équation (4.20). $(.,.)_{A_{R^k,\Gamma_*^k}}$ est symétrique et elliptique. $(.,.)_{f_{R^k}}$ est symétrique et définie positive, puisqu'on a supposé que la section de fission est strictement positive.

**Lemme 4.3** *En choisissant les $\{\varphi_i^k\}_{i\geq 1}$ avec l'orthonormalisation donnée dans le lemme 4.2, on a, $\forall\psi \in H^1_{0,\Gamma^k}(R^k)$ :*

$$\|\psi\|^2_{A_{R^k,\Gamma_*^k}} = \sum_{i=1}^{\infty}\lambda_i^k{\alpha_i^k}^2. \tag{4.28}$$

**Preuve** On a :

$$\begin{aligned}\|\psi\|^2_{A_{R^k,\Gamma_*^k}} &= \int_{R^k} D(\overrightarrow{\nabla}\psi)^2 + \int_{R^k}\sigma_a\psi^2 + \alpha\int_{\Gamma_*^k}\psi^2\\ &= \int_{R^k} D\left(\overrightarrow{\nabla}\left(\sum_{i=1}^{\infty}\alpha_i^k\varphi_i^k\right)\right)^2 + \int_{R^k}\sigma_a\left(\sum_{i=1}^{\infty}\alpha_i^k\varphi_i^k\right)^2 + \alpha\int_{\Gamma_*^k}\left(\sum_{i=1}^{\infty}\alpha_i^k\varphi_i^k\right)^2\\ &= \sum_{i=1}^{\infty}\int_{R^k} D{\alpha_i^k}^2\overrightarrow{\nabla}{\varphi_i^k}^2 + \sum_{i=1}^{\infty}\int_{R^k}\sigma_a{\alpha_i^k}^2{\varphi_i^k}^2 + \sum_{i=1}^{\infty}\alpha\int_{\Gamma_*^k}{\alpha_i^k}^2{\varphi_i^k}^2\\ &+ \sum_{i\neq j}\int_{R^k} D\beta_{ij}\alpha_i^k\alpha_j^k\overrightarrow{\nabla}\varphi_i^k\cdot\overrightarrow{\nabla}\varphi_j^k + \sum_{i\neq j}\int_{R^k}\sigma_a\beta_{ij}\alpha_i^k\alpha_j^k\varphi_i^k\varphi_j^k + \sum_{i\neq j}\alpha\int_{\Gamma_*^k}\beta_{ij}\alpha_i^k\alpha_j^k\varphi_i^k\varphi_j^k.\end{aligned}$$

Or,

$$\begin{aligned}
&\sum_{i\neq j}\int_{R^k} D\beta_{ij}\alpha_i^k\alpha_j^k\overrightarrow{\nabla}\varphi_i^k\cdot\overrightarrow{\nabla}\varphi_j^k+\sum_{i\neq j}\int_{R^k}\sigma_a\beta_{ij}\alpha_i^k\alpha_j^k\varphi_i^k\varphi_j^k+\sum_{i\neq j}\alpha\int_{\Gamma_*^k}\beta_{ij}\alpha_i^k\alpha_j^k\varphi_i^k\varphi_j^k\\
&=\sum_{i\neq j}\gamma_{ij}\left(\int_{R^k} D\overrightarrow{\nabla}\varphi_i^k\cdot\overrightarrow{\nabla}\varphi_j^k+\int_{R^k}\sigma_a\varphi_i^k\varphi_j^k+\alpha\int_{\Gamma_*^k}\varphi_i^k\varphi_j^k\right)\\
&=\sum_{i\neq j}\gamma_{ij}\lambda_i^k\int_{R^k}\sigma_f\varphi_i^k\varphi_j^k\\
&=0,
\end{aligned}$$

d'après la propriété d'orthonormalité de la famille $\{\varphi_i^k\}_{i\geq 1}$. Donc

$$\begin{aligned}
\|\psi\|^2_{A_{R^k,\Gamma_*^k}} &=\sum_{i=1}^{\infty}\int_{R^k} D\alpha_i^{k\,2}\overrightarrow{\nabla}\varphi_i^{k\,2}+\sum_{i=1}^{\infty}\int_{R^k}\sigma_a\alpha_i^{k\,2}\varphi_i^{k\,2}+\sum_{i=1}^{\infty}\alpha\alpha_i^{k\,2}\int_{\Gamma_*^k}\varphi_i^{k\,2}\\
&=\sum_{i=1}^{\infty}\alpha_i^{k\,2}\left(\int_{R^k} D\left(\overrightarrow{\nabla}\varphi_i^k\right)^2+\int_{R^k}\sigma_a\varphi_i^{k\,2}+\alpha\int_{\Gamma_*^k}\varphi_i^{k\,2}\right)\\
&=\sum_{i=1}^{\infty}\lambda_i^k\alpha_i^{k\,2}\int_{R^k}\sigma_f\varphi_i^{k\,2}\\
&=\sum_{i=1}^{\infty}\lambda_i^k\alpha_i^{k\,2}.
\end{aligned}$$

**Lemme 4.4** *On suppose que* $\psi\in H^1_{0,\Gamma^k}(R^k)$ *et que* $\forall p\geq 1,\left(\frac{A_{R^k,\Gamma_*^k}}{\sigma_f}\right)^p(\psi)\in H^1_{0,\Gamma^k}(R^k)$. *Alors les* $\alpha_i^k$ *définis au lemme 4.2 sont tels que :*

$$\alpha_i^k=\left(\frac{1}{\lambda_i^k}\right)^p\left(\left(\frac{A_{R^k,\Gamma_*^k}}{\sigma_f}\right)^p(\psi),\varphi_i^k\right)_{f_{R^k}}\qquad\forall p\geq 0. \tag{4.29}$$

**Preuve** La démonstration est récursive, basée sur le fait que $\varphi_i^k$ est solution du problème (4.24).

$$\begin{aligned}
\alpha_i^k &=\left(\psi,\varphi_i^k\right)_{f_{R^k}}\\
&=\frac{1}{\lambda_i^k}\left(\psi,\varphi_i^k\right)_{A_{R^k,\Gamma_*^k}}\\
&=\frac{1}{\lambda_i^k}\left(\frac{A_{R^k,\Gamma_*^k}}{\sigma_f}\psi,\varphi_i^k\right)_{f_{R^k}}.
\end{aligned}$$

Supposons que la propriété (4.29) est vraie à l'ordre n. Alors :

$$\begin{aligned}
\alpha_i^k &=\left(\frac{1}{\lambda_i^k}\right)^n\left(\left(\frac{A_{R^k,\Gamma_*^k}}{\sigma_f}\right)^n\psi,\varphi_i^k\right)_{f_{R^k}}\\
&=\left(\frac{1}{\lambda_i^k}\right)^{n+1}\left(\left(\frac{A_{R^k,\Gamma_*^k}}{\sigma_f}\right)^n\psi,\varphi_i^k\right)_{A_{R^k,\Gamma_*^k}}\\
&=\left(\frac{1}{\lambda_i^k}\right)^{n+1}\left(\left(\frac{A_{R^k,\Gamma_*^k}}{\sigma_f}\right)^{n+1}\psi,\varphi_i^k\right)_{f_{R^k}}.
\end{aligned}$$

La propriété est donc vraie $\forall p\geq 0$.

### 4.3.3 L'estimation d'erreur

Notre but est d'étudier la convergence de la méthode de synthèse modale appliquée à la résolution du problème (4.17), et d'estimer les erreurs entre les solutions approchées et les solutions exactes en les majorant par une quantité qui décroît lorsque le nombre de modes locaux utilisés dans la base d'approximation augmente.

Nous allons montrer que sur chaque sous-domaine $R^k$, les solutions du problème global (4.18) sont bien approchées par des fonctions de l'espace engendré par les $N^k$ premières solutions locales du problème (4.24) prolongées par zéro sur $R \setminus R^k$. De plus, lorsque $N^k \to +\infty$, nous allons voir que les écarts entre les solutions approchées et exactes tendent vers zéro.

Avant de donner notre résultat sur l'estimation de l'erreur, nous rappelons un théorème dont on pourra trouver la démonstration dans [Babuska et Osborn, 1991].

**Théorème 4.1** *Soient les solutions $(u_i \in W, \lambda_i \in \mathbb{R})_{i\geq 1}$ d'un problème du type :*

$$a(u_i, v) = \lambda_i b(u_i, v), \quad \forall v \in W. \tag{4.30}$$

*On suppose que $W$ est un espace de Hilbert muni d'une norme $\|.\|_W$, $a$ est une forme bilinéaire symétrique elliptique, $b$ est une forme bilinéaire symétrique définie positive.*

*Soit $W_\delta \subset W$ une famille d'espaces de dimension finie $\delta$ satisfaisant $\lim_{\delta\to\infty} \inf_{v\in W_\delta} \|u - v\|_W = 0, \forall u \in W$.*

*Soient $(u_{i,\delta}, \lambda_{i,\delta})$ les solutions du problème approché :*

$$a(u_{i,\delta}, v) = \lambda_{i,\delta} b(u_{i,\delta}, v), \quad \forall v \in W_\delta. \tag{4.31}$$

*Alors il existe une constante $C$ telle que :*

$$\lambda_i \leq \lambda_{i,\delta} \leq \lambda_i + C\epsilon_\delta^2(\lambda_i), \tag{4.32}$$

*où, pour chaque valeur propre $\lambda$ du problème (4.30) associée à l'espace propre $V(\lambda)$, on a :*

$$\epsilon_\delta(\lambda) = \max_{v\in V(\lambda), \|v\|_W=1} \inf_{v_\delta \in W_\delta} \|v - v_\delta\|_W . \tag{4.33}$$

Ce théorème s'applique à notre cas, avec $a(.,.) = (.,.)_{A_{R,\Gamma}}$ et $b(.,.) = (.,.)_{f_R}$. L'espace d'approximation de la synthèse modale est constitué de l'ensemble des solutions des problèmes locaux (4.24) sur tous les sous-domaines. Ces solutions sont prolongées par zéro sur $R \setminus R^k$ de façon à avoir des fonctions définies sur $R$. Soit un entier $l$ tel que $l \leq \inf_{1\leq k\leq K} N^k$. On définit l'espace d'approximation $X_{\delta,l} = Vect\{\tilde{\varphi}_i^k\}_{1\leq i\leq l}^{1\leq k\leq K}$, le $\tilde{}$ désignant le prolongement par zéro. Le prolongement par zéro des $\varphi_i^k$ assure la continuité des $\tilde{\varphi}_i^k$ sur $\Gamma^k$ car $\varphi_i^k \in H^1_{0,\Gamma^k}(R^k)$. C'est une condition suffisante pour montrer que $X_{\delta,l} \subset H^1(R)$.

Les espaces du théorème 4.1 sont dans notre cas $W = H^1(R)$ et $W_\delta = X_{\delta,l}$ Nous pouvons maintenant énoncer le théorème principal qui fournit une estimation de l'erreur, et prouve la convergence d'ordre infini de la méthode de synthèse modale pour le problème de la diffusion (4.17).

**Théorème 4.2** *Soit $V(\lambda)$ l'espace propre associé à la valeur propre $\lambda$ du problème (4.18). On définit :*

$$M_\lambda(p) = \max_{v\in V(\lambda), \|v\|_{H^1(R)}=1} \min_{\{\chi^k\}_{1\leq k\leq K}\in\chi_R} \max_{1\leq k\leq K} \left\| \left( \frac{A_{R^k,\Gamma_*^k}}{\sigma_f} \right)^p (\chi^k v) \right\|_{A_{R^k,\Gamma_*^k}} . \tag{4.34}$$

*Il existe une constante $C > 0$ telle que $\forall p > 0$, pour tout ensemble d'entiers $\{N^k\}_{1\leq k\leq K}$,*

$$\epsilon_\delta(\lambda) \leq C M_\lambda(p) \Lambda^{-p}, \tag{4.35}$$

*avec* $\Lambda = \inf_{1\leq k\leq K} \lambda^k_{N^k+1}$

**Preuve** On introduit un projecteur $P_{\delta,l}$ définit par :

$$\begin{array}{rccl} P_{\delta,l}: & H^1(R) & \to & X_{\delta,l} \\ & P_{\delta,l}(\psi) & = & \displaystyle\sum_{k=1}^{K} P^k_{\delta,l}(\chi^k\psi), \end{array} \tag{4.36}$$

avec $P^k_{\delta,l}$ défini comme le projecteur orthogonal de $H^1(R)$ sur $X^k_{\delta,l} = Vect\{\tilde{\varphi}^k_i\}_{1\leq i\leq l}$ pour le produit scalaire $(.,.)_{A_{R,\Gamma}}$. Pour toute fonction propre $\varphi_i$ solution de (4.18), on a la majoration suivante :

$$\begin{aligned} \inf_{\varphi_\delta\in X_{\delta,l}} \|\varphi_i - \varphi_\delta\|_{A_{R,\Gamma}} &\leq \|\varphi_i - P_{\delta,l}\varphi_i\|_{A_{R,\Gamma}} \\ &\leq \left\| \left(\sum_{k=1}^{K} \chi^k\varphi_i\right) - \left(\sum_{k=1}^{K} P^k_{\delta,l}(\chi^k\varphi_i)\right) \right\|_{A_{R,\Gamma}} \\ &\leq \sum_{k=1}^{K} \left\| \chi^k\varphi_i - P^k_{\delta,l}(\chi^k\varphi_i) \right\|_{A_{R,\Gamma}} \\ &\leq \sum_{k=1}^{K} \inf_{\psi\in X^k_{\delta,l}} \left\| \chi^k\varphi_i - \psi \right\|_{A_{R^k,\Gamma^k_*}}. \end{aligned} \tag{4.37}$$

Supposons que $\forall p \geq 0, \left(\frac{A_{R^k,\Gamma^k_*}}{\sigma_f}\right)^p(\psi) \in H^1_{0,\Gamma^k}(R^k)$. D'après les lemmes 4.3 et 4.4, on a $\forall p \geq 0$ :

$$\begin{aligned} \left\| \psi - P^k_{\delta,l}(\psi) \right\|^2_{A_{R^k,\Gamma^k_*}} &= \sum_{i=l+1}^{+\infty} \lambda^k_i \alpha^{k\,2}_i \\ &= \sum_{i=l+1}^{\infty} \lambda^k_i \left(\frac{1}{\lambda^k_i}\right)^{2p} \left( \left(\frac{A_{R^k,\Gamma^k_*}}{\sigma_f}\right)^p \psi, \varphi^k_i \right)^2_{f_{R^k}} \\ &\leq \left(\frac{1}{\lambda^k_{l+1}}\right)^{2p} \sum_{i=l+1}^{\infty} \lambda^k_i \left( \left(\frac{A_{R^k,\Gamma^k_*}}{\sigma_f}\right)^p \psi, \varphi^k_i \right)^2_{f_{R^k}} \\ &\leq \left(\frac{1}{\lambda^k_{l+1}}\right)^{2p} \left\| \left(\frac{A_{R^k,\Gamma^k_*}}{\sigma_f}\right)^p \psi \right\|^2_{A_{R^k,\Gamma^k_*}}, \end{aligned}$$

d'après le lemme 4.3. Ce résultat peut-être appliqué à $\psi = \chi^k\varphi_i$. En effet, d'après le lemme 4.1, et en utilisant la propriété (4.22) des fonctions $\chi^k \in C^\infty(R)$, on a :

$$\left(\frac{A_{R^k,\Gamma^k_*}}{\sigma_f}\right)^p (\chi^k\varphi_i) \in H^1_{0,\Gamma^k}(R^k), \quad \forall p \geq 0. \tag{4.38}$$

On obtient donc, $\forall p \geq 0$ :

$$\begin{aligned} \left\| \chi^k\varphi_i - P^k_{\delta,l}(\chi^k\varphi_i) \right\|^2_{A_{R^k,\Gamma^k_*}} &= \inf_{\varphi_\delta\in X_{\delta,l}} \left\| \chi^k\varphi_i - \varphi_\delta \right\|^2_{A_{R^k,\Gamma^k_*}} \\ &\leq \left(\frac{1}{\lambda^k_{l+1}}\right)^{2p} \left\| \left(\frac{A_{R^k,\Gamma^k_*}}{\sigma_f}\right)^p (\chi^k\varphi_i) \right\|^2_{A_{R^k,\Gamma^k_*}}. \end{aligned}$$

En injectant cette relation dans la dernière inégalité de (4.37), on obtient bien la relation (4.35).

### 4.3.4 Commentaires sur la démonstration

On a donc montré une estimation d'erreur pour l'approximation des solutions du problème à valeur propre de la diffusion par la méthode de synthèse modale. La démonstration est faite dans le cadre de la formulation primale, avec des conditions de Robin aux bords de $R$.

Plusieurs remarques sont à faire en ce qui concerne l'extension de cette démonstration à des cas plus généraux. La première est que l'application à des conditions de flux nul aux bords implique que (4.38) soit dans $H_0^1(R^k)$. Or il ne semble pas évident à priori de montrer que (4.38) s'annule sur $\partial R$ lorsque $\partial R^k \cap \partial R \neq \emptyset$. Cette démonstration est faite dans [Charpentier *et al.*, 1995] pour l'opérateur Laplacien.

La deuxième remarque est que l'hypothèse $\sigma_f > 0$ est nécessaire pour que $(.,.)_{f_R}$ soit un produit scalaire. C'est donc une hypothèse nécessaire pour que les modes propres des problèmes locaux forment une base complète de $H^1_{0,\Gamma^k}(R^k)$. La démonstration sans cette hypothèse implique un raisonnement dans des espaces différents :

- l'espace $H^1(R)$ est remplacé par l'espace $X$ orthogonal pour le produit scalaire $(.,.)_{A_{R,\Gamma}}$ à l'espace $Y$ définit par : $Y = \{\varphi \in H^1(R), \varphi = 0 \text{ sur } R^*\}$, avec $R^* = \{x \in R, \sigma_f(x) \neq 0\}$ ;
- l'espace $H_0^1(R^k)$ est remplacé par l'espace $X^k$ orthogonal pour le produit scalaire $(.,.)_{A_{R^k,\Gamma^k_*}}$ à l'espace $Y^k$ définit par : $Y^k = \{\varphi \in H^1_{0,\Gamma^k}(R^k), \varphi = 0 \text{ sur } R^{k*}\}$, avec $R^{k*} = \{x \in R^k, \sigma_f(x) \neq 0\}$.

On a alors $(.,.)_{f_X}$ et $(.,.)_{f_{X^k}}$ qui sont des produits scalaires. Mais l'appartenance de (4.38) à $X^k$ est à priori fausse, car la propriété d'orthogonalité n'a pas de raison d'être vérifiée.

La troisième remarque est que la démonstration repose sur l'hypothèse d'un opérateur $A_{R,\Gamma}$ autoadjoint. On ne peut donc pas la généraliser à l'équation de la diffusion critique multigroupe.

On voit donc que le domaine de validité de l'estimation d'erreur (4.35) est restreint, et ne correspond pas aux cas réels des cœurs de réacteur. Cependant, cela nous donne une idée de la convergence de la méthode, et conforte notre démarche d'application de la méthode de synthèse modale au problème de la diffusion des neutrons.

## 4.4 Tests de la méthode CMS en 1D

Afin de valider l'approche de la synthèse modale pour la formulation mixte duale du problème de la diffusion des neutrons, nous présentons des tests numériques sur un cas simple : une dimension d'espace et un groupe d'énergie. Dans ce cas l'opérateur de diffusion est autoadjoint. Le domaine de calcul est fortement hétérogène de manière à reproduire les variations discontinues des section efficaces dans un cœur de réacteur. A ce sujet, on trouvera dans [Charpentier *et al.*, 1998] une étude de la méthode de synthèse modale avec des sous-domaines recouvrants dans le cas d'un opérateur auto-adjoint avec des coefficients discontinues.

Un programme a été réalisé en *Scilab*. En plus de la méthode de synthèse modale, un solveur de type MINOS a été implémenté afin d'avoir une solution de "référence" obtenue par un calcul direct.

### 4.4.1 La géométrie 1D

La géométrie d'un cœur peut difficilement être simplifiée pour obtenir un modèle 1D. Néanmoins, nous avons choisi une géométrie à une dimension d'espace qui permet de retrouver certaines caractéristiques d'un cœur tel que celui du REP (voir section 1.1.1).

Notre modèle 1D de cœur mesure 340cm, et est divisé en 17 assemblages. Deux assemblages $A1$ et $A2$ sont utilisés alternativement, comme le montre la figure 4.1. Chaque assemblage est lui-même composé de 17 crayons. Nous avons fixé des valeurs constantes des sections efficaces $D, \sigma_a, \sigma_f$ sur chaque crayon. Deux jeux de valeurs des sections correspondant à deux crayons différents s'alternent sur chaque assemblage, afin de simuler un milieu hétérogène. Le tableau 4.1 donne les valeurs des sections sur les crayons des deux assemblages $A1$ et $A2$. On remarque que la section de fission $\sigma_f$ est choisie nulle sur certains crayons. Outre le fait que cette configuration correspond à des milieux réalistes (modérateur, caloporteur, réflecteur...), cela permet de tester la validité de la méthode CMS sur des cas où l'hypothèse d'une section de fission strictement positive nécessaire à l'obtention de l'estimation d'erreur (4.35) n'est pas vérifiée.

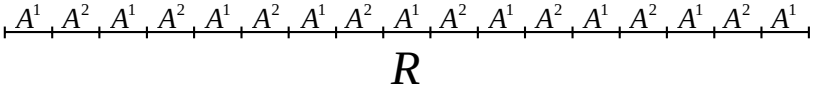



FIG. 4.1: Géométrie du domaine 1D, composé de deux assemblages $A1$ et $A2$.

TAB. 4.1: Valeurs des sections sur les quatre crayons des deux assemblages de la géométrie 1D

| | $A1$, crayon 1 | $A1$, crayon 2 | $A2$, crayon 1 | $A2$, crayon 2 |
|---|---|---|---|---|
| $\sigma_a$ | $9.6 \times 10^{-3}$ | $1.8 \times 10^{-2}$ | $8 \times 10^{-3}$ | $1.3 \times 10^{-2}$ |
| $D$ | 1 | 2 | 1 | 2 |
| $\sigma_f$ | $2 \times 10^{-2}$ | 0 | $2.5 \times 10^{-2}$ | 0 |

Un calcul direct utilisant 4 mailles par crayon et l'élément $RT_0$ est pris comme référence. La valeur propre fondamentale est $k_{eff} = 1,008238$ et le flux est représenté sur la figure 4.2. Comme on peut le voir, le flux ne change pas de signe. On constate des variations à plusieurs niveaux. La forme macroscopique est donnée par un sinus, et on observe des variations assez fortes à l'échelle des assemblages, ainsi que des variations plus petites et plus fréquentes à l'échelle des crayons.

### 4.4.2 Analyse de l'erreur d'approximation en fonction des paramètres de la méthode CMS

Les tests que nous allons présenter ont pour but de mesurer l'impact des différents paramètres de la méthode CMS sur la qualité de l'approximation. Ainsi, nous allons faire varier la taille des sous-domaines, leurs configurations, la taille de leurs recouvrements, le nombre de modes de flux sur chaque sous-domaine et le nombre de modes de courant. L'erreur d'approximation est mesurée par rapport à la solution de référence. Les erreurs relatives sur le $k_{eff}$ et sur le flux pour les normes $L^2, L^\infty$ et $H^1$ sont données. A priori on s'attend à vérifier deux phénomènes ([Charpentier *et al.*, 1996]) : l'erreur doit diminuer lorsque le nombre

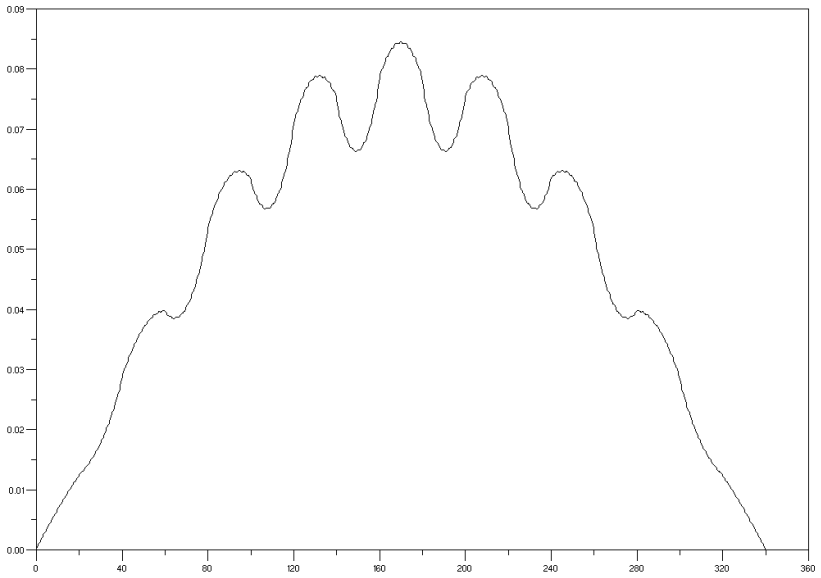



FIG. 4.2: Représentation du flux sur la géométrie 1D. $k_{eff} = 1,008238$.

de modes augmente (les espaces d'approximation sont plus grands), et lorsque les tailles des recouvrements sont plus grandes.

La première décomposition de domaine testée utilise trois sous-domaines de taille égale à sept assemblages (figure 4.3). Les sous-domaines se recouvrent deux à deux, avec un recouvrement de deux assemblages. Le nombre de modes choisi est dans un premier temps le même pour tous les sous-domaines, pour le courant et pour le flux : $N_p^k = N_\varphi^k, \forall k$. La figure 4.4a représente l'erreur relative en fonction du nombre de modes choisi. Il apparaît sur cette figure que l'erreur est parfois plus grande avec un nombre de modes plus élevé. Comme nous l'avons évoqué dans la section précédente, il est nécessaire d'avoir un espace de courant suffisamment grand pour avoir une bonne approximation, ce qui n'est pas le cas lorsque le nombre de modes de courant et le nombre de modes de flux sont égaux. On constate que pour $N_p^k = N_\varphi^k + 1$ (figure 4.4b) et pour $N_p^k = N_\varphi^k + 2$ (figure 4.4c), l'approximation s'améliore nettement. Il est à noter que lorsque $N_\varphi^k > 15$, le conditionnement du système linéaire à résoudre est très mauvais, car un ou plusieurs modes de flux ou de courant sont proches d'une combinaison linéaire d'autres modes. Ceci pourrait être corrigé en retirant de la base ces modes presque liés, mais nous n'avons pas implémenté un tel algorithme.

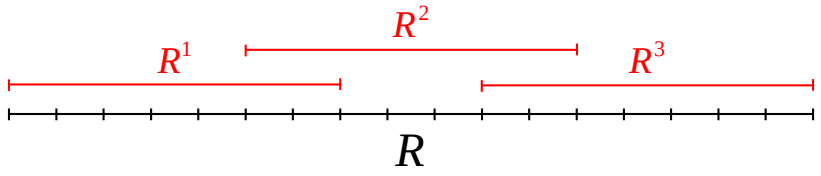
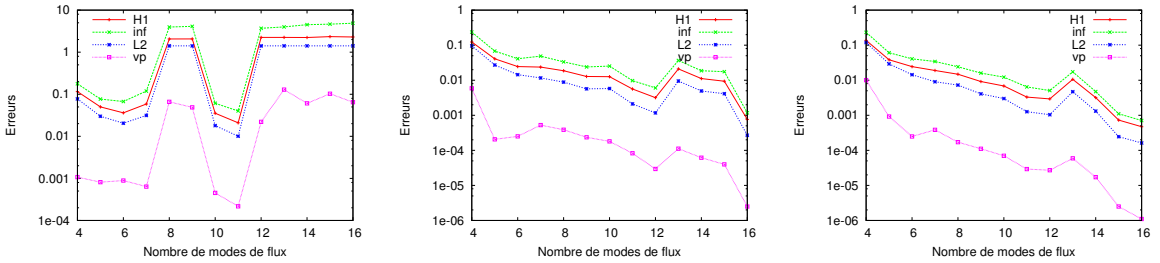



FIG. 4.3: Décomposition en trois sous-domaines.

Un bon compromis semble ici de prendre $N_p^k = N_\varphi^k + 1$. Certes l'approximation est meilleure pour $N_p^k = N_\varphi^k + 2$, mais les deux modes de flux supplémentaires calculés sur chaque sous-




(a) $\forall k, N_p^k = N_\varphi^k$ (b) $\forall k, N_p^k = N_\varphi^k + 1$ (c) $\forall k, N_p = N_\varphi^k + 2$

FIG. 4.4: Erreurs relatives d'approximation pour une décomposition en trois sous-domaines.

domaine représentent un surcoût de la méthode puisqu'ils ne servent pas de fonctions de base pour l'espace de flux.

La décomposition de domaine suggérée "naturellement" par la géométrie du domaine utilise des sous-domaines correspondant aux assemblages, agrandis de chaque côté de manière à assurer le recouvrement (figure 4.5). On choisit donc une décomposition en 17 sous-domaines, chaque sous-domaine étant un assemblage auquel on ajoute un petit recouvrement de 3 crayons de chaque côté. La figure 4.6a présente l'erreur relative d'approximation en fonction du nombre de modes, avec $N_p^k = N_\varphi^k + 1, \forall k$. Afin d'évaluer l'influence de la taille du recouvrement, on donne sur la figure 4.6b les résultats correspondant à la même décomposition de domaine, mais avec un recouvrement plus large de 7 crayons de chaque côté.

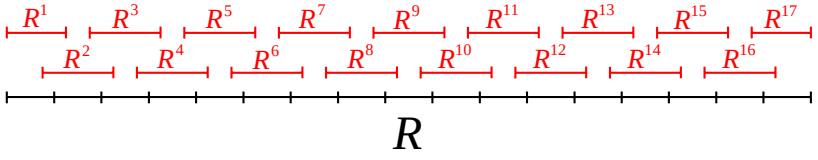
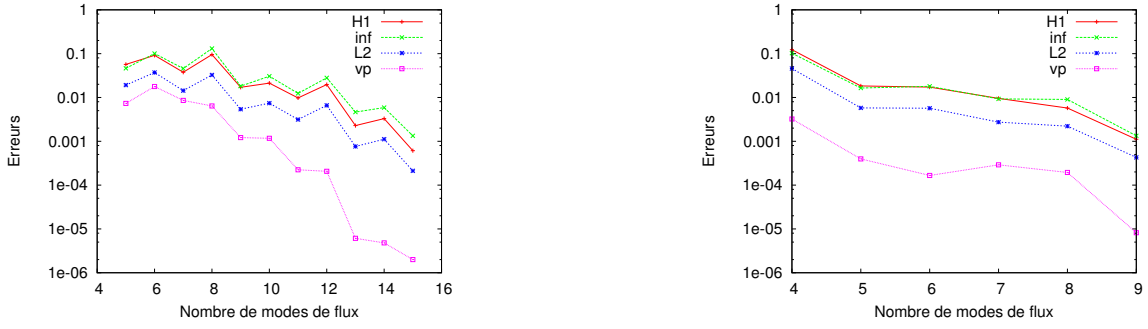



FIG. 4.5: Décomposition en 17 sous-domaines.




(a) Recouvrement de 3 crayons (b) Recouvrement de 7 crayons

FIG. 4.6: Erreurs relatives d'approximation pour une décomposition en 17 sous-domaines. $N_p^k = N_\varphi^k + 1, \forall k$.

L'approximation est meilleure avec un recouvrement plus grand : moins de modes sont nécessaires pour avoir une erreur du même ordre. L'augmentation du nombre de sous-domaines

ne semble pas pénalisante pour la précision de la méthode lorsque l'on compare les graphiques 4.6b et 4.4b. Cela s'explique sans doute par le fait que plus le nombre de sous-domaines est élevé, moins les modes locaux approchent correctement les allures globales du flux et du courant, mais plus les dimensions des espaces d'approximation pour la méthode de synthèse modale sont grandes. Quant aux variations à l'échelle du crayon, elles sont suffisamment localisées pour être prises en compte par tous les modes locaux, quelque soit la tailles des sous-domaines.

Nous souhaitons maintenant tester une approche plus "physique" pour la décomposition de domaine, qui prenne en compte le fait qu'à priori le flux est plus plat et le courant est plus faible au milieu des assemblages que sur leurs bords. Les conditions de courant nul imposées sur les bords internes des sous-domaines seront donc à priori mieux vérifiées si on choisit les bords des sous-domaines au milieu des assemblages. De plus, les flux étant discontinus aux bords des sous-domaines à cause du prolongement des modes locaux par zéro, on souhaite que ces bords se touchent deux à deux. On adopte donc une décomposition du domaine avec un recouvrement total, et avec des sous-domaines dont les bords correspondent aux milieux des assemblages. Leur taille est de deux assemblages, sauf pour ceux aux bords du domaine (figure 4.7). Là aussi on fait varier le nombre de modes, avec $N_p^k = N_\varphi^k + 1, \forall k$. Comme le montre le graphique 4.8 , l'approximation est excellente, même avec un petit nombre de modes. Par contre, comme les sous-domaines se recouvrent totalement, le système linéaire est vite mal conditionné lorsque l'on augmente le nombre de modes (ici pour $N_\varphi^k \geq 4$).

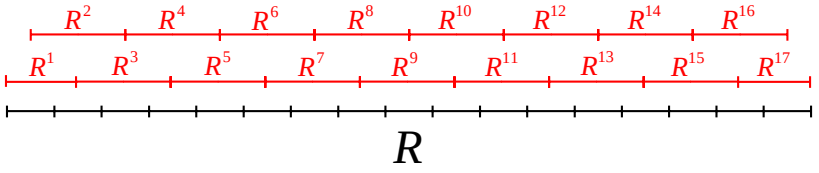



FIG. 4.7: Décomposition en 17 sous-domaines totalement recouvrants.

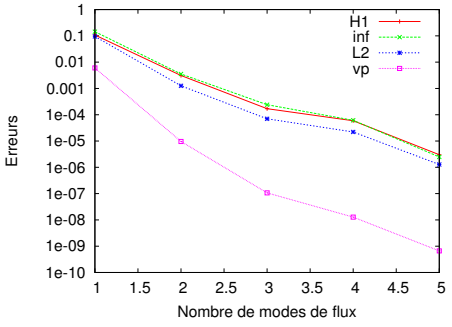



FIG. 4.8: Erreurs relatives d'approximation pour une décomposition en 17 sous-domaines totalement recouvrants. $N_p^k = N_\varphi^k + 1, \forall k$.

*Nous avons présenté dans ce chapitre une méthode de synthèse modale avec recouvrement pour des problèmes sous forme mixte. Nous l'avons appliquée à la formulation mixte duale du problème critique de la diffusion. Avant de réaliser les premiers tests en dimension 2 ou 3, il est nécessaire de tirer quelques conclusions de l'étude théorique de la méthode CMS, et des tests en 1D que nous venons de présenter. En effet il en ressort plusieurs contraintes à prendre en compte.*

*La première est la nécessité d'avoir plus de modes de courant que de flux pour avoir une approximation correcte de la solution. Mais on souhaite que cette différence ne soit pas trop grande pour ne pas à avoir à calculer trop de modes de flux inutiles par la suite, car les calculs locaux fournissent autant de modes de flux que de courant. En 1D il apparaît qu'un mode de courant supplémentaire par sous-domaine est suffisant, mais ça n'est à priori pas le cas en dimension supérieure.*

*La deuxième contrainte vient de la décomposition du domaine. Il semble qu'une configuration du type 4.7, où les sous-domaines se recouvrent totalement et ont pour bords les milieux des assemblages, soit idéale. Il faudra donc essayer de généraliser cette configuration en 2D et en 3D afin d'obtenir une bonne approximation en calculant peu de modes locaux.*

*La troisième contrainte vient du fait qu'il ne faut pas utiliser trop de modes, car le système linéaire est alors très mal conditionné. Un remède serait d'ajouter un algorithme pour éliminer les modes numériquement dépendants des autres.*

*La méthode CMS est donc très sensible à trois paramètres : le nombre de modes de flux, le nombre de modes de courant et la décomposition de domaine. C'est un handicap de la méthode vis-à-vis de l'utilisateur, car le choix de ces paramètres est difficile à faire à priori. Cependant, les tests numériques montrent que le respect de ces contraintes permet d'obtenir une approximation très précise de la solution. Mieux, cette qualité d'approximation est obtenue avec une section de fission qui s'annule sur un grand nombre de crayons. Or rien ne garantissait à priori la précision de la méthode CMS dans ce cas, puisque l'estimation d'erreur (4.35) n'a été montrée que sous l'hypothèse d'une section de fission strictement positive.*

*Nous allons maintenant présenter l'application de la méthode CMS à des cas industriels "réalistes", en 2D et en 3D. L'influence des paramètres de la méthode sur la qualité d'approximation pourra ainsi être étudiée sur des géométries plus complexes.*

# Chapitre 5

# Applications numériques et parallélisation de la méthode CMS

---

*Dans le chapitre précédent nous avons introduit la méthode CMS : c'est une méthode de synthèse modale avec des sous-domaines recouvrants pour la formulation mixte duale du problème critique de la diffusion. Les tests 1D présentés ont montré la qualité d'approximation qu'apporte la méthode CMS, même avec un petit nombre de modes. Mais la sensibilité de la méthode à certains paramètres a également été mise en évidence.*

*Afin de valider la méthode CMS sur des géométries 2D et 3D complexes et hétérogènes, un programme en C++ a été développé dans le cadre du projet APOLLO3/DESCARTES. Après avoir exposé la programmation de la méthode, nous présentons des tests sur le cœur du REP en 2D et 3D. La qualité de l'approximation est étudiée, ainsi que la dépendance aux différents paramètres. Puis la parallélisation de la méthode est abordée, en détaillant les étapes de calcul sur chaque processeur.*

---

## 5.1 La méthode CMS dans APOLLO3/DESCARTES

Avant le calcul de cœur proprement dit, nous avons vu que le domaine, les sections efficaces et la géométrie sont définis. Or notre approche utilise une décomposition du domaine en plusieurs sous-domaines recouvrants. Nous avons en fait gardé le domaine et les sections communs à tous les sous-domaines, ces derniers étant définis par une géométrie individuelle. L'intérêt de garder un domaine et des sections globales est notamment de pouvoir reconstruire de manière simple les flux et la puissance sur le cœur complet.

Contrairement aux tests en une dimension présentés à la section 4.4, nous prenons en compte plusieurs groupes d'énergie. Contrairement à l'approche développée dans [Pinchedez, 1999], nous faisons le choix de calculer les modes locaux avec l'opérateur non-autoadjoint de la diffusion multigroupe. L'obtention d'une base orthogonale de vecteurs propres n'est donc pas assurée, bien que la conjecture de réalité du spectre laisse entrevoir la possibilité de la construction numérique d'une telle base. Cependant, l'orthogonalité de la base n'est pas nécessaire pour la méthode CMS. En effet, il suffit pour engendrer les espaces $V_\delta$ et $W_\delta$ d'avoir un ensemble de vecteurs propres de flux et de courant linéairement indépendants.

Le calcul avec $N_g$ groupes d'énergie est géré de la même manière que dans l'algorithme de MINOS, mais quelques particularités liées à la méthode CMS sont à prendre en compte. Nous avons vu à la section 4.2 que la méthode de synthèse modale se décompose en trois étapes.

### 5.1.1 Calcul des modes locaux

La première étape est le calcul des modes locaux sur chaque sous-domaine réalisé par MINOS. La résolution locale se fait de manière multigroupe : on obtient sur chaque sous-domaine le flux et le courant pour chaque groupe. Mais seul le mode fondamental est calculé par la méthode des puissances, et nous avons besoin de modes d'ordres supérieurs dans la méthode CMS. Plutôt que d'utiliser une autre méthode, qui nécessiterait des changements importants du solveur MINOS, nous avons modifié la méthode des puissances.

Dans le cas autoadjoint à un seul groupe d'énergie, on a vu que si $\varphi_i$ et $\varphi_j$ sont solutions propres du problème de la diffusion critique (4.17), alors si $i \neq j$, $\int_\Omega S_i \varphi_j = 0$, avec $S_i = \sigma_f \varphi_i$ (voir le lemme 4.2). Mais cette relation n'est plus vraie dans le cas multigroupe non autoadjoint, et il faudrait faire intervenir le flux adjoint pour avoir une relation d'orthogonalité. Or le calcul du flux adjoint est aussi coûteux que le calcul du flux lui-même. Nous avons donc choisi d'utiliser une relation d'orthogonalité sur les sources à priori fausse : $\int_\Omega S_i S_j = 0, i \neq j$. Mais pour tous les tests réalisés en 1D, cette relation fournit une excellente approximation des premiers modes propres.

L'algorithme de la puissance modifié 3 permet de calculer les $N^{mode}$ premiers modes. Même si cet algorithme ne garantit pas la qualité d'approximation des modes supplémentaires, il permet néanmoins d'obtenir des fonctions de base pour la méthode CMS, qui décrivent correctement l'allure locale du flux et du courant. De plus il présente l'avantage de ne nécessiter que des petites modifications du solveur MINOS. Il réclame par contre le stockage des vecteurs sources associés à chaque mode, de taille égale au vecteur de flux.

L'implémentation d'une méthode robuste de calcul des modes propres dans le solveur MINOS, telle que l'algorithme de Jacobi-Davidson ([Verdu *et al.*, 2005]), permettrait d'obtenir les solutions locales plus rapidement et avec une plus grande fiabilité.

**Algorithme 3** Algorithme des puissances pour le calcul de plusieurs modes

{Boucle sur les modes}
**pour** $i = 1$ à $N^{mode}$ **faire**
Initialisations du vecteur $U_i^1$ et de la valeur propre $k_i^1$
Calcul de la source initiale normalisée $\bar{S}_i^1 = \dfrac{FU_i^1}{k_i^1}$
Orthogonalisation de la source initiale par rapport aux source précédentes :
**pour** $j = 1$ à $i - 1$ **faire**
$\bar{S}_i^1 = \bar{S}_i^1 - \dfrac{< \bar{S}_i^1, \bar{S}_j >}{< \bar{S}_j, \bar{S}_j >} \bar{S}_j$
**fin pour**

{Itérations externes}
**pour** $n = 1$ à $N^{max}$ **faire**

Résolution de $\mathcal{H}U_i^{n+1} = \bar{S}_i^n$
Mise à jour de la source : $S_i^{n+1} = FU_i^{n+1}$
Orthogonalisation de la source par rapport aux source précédentes :
**pour** $j = 1$ à $i - 1$ **faire**
$S_i^{n+1} = S_i^{n+1} - \dfrac{< \bar{S}_i^{n+1}, \bar{S}_j >}{< \bar{S}_j, \bar{S}_j >} \bar{S}_j$
**fin pour**
Calcul de la nouvelle valeur propre : $k_i^{n+1} = \dfrac{\left\| S_i^{n+1} \right\|_2^2}{(S_i^{n+1}, \bar{S}_i^n)}$
Calcul de la source normalisée : $\bar{S}_i^{n+1} = \dfrac{S_i^{n+1}}{k_i^{n+1}}$
Mise à jour de l'erreur : $Err = \dfrac{\left\| \bar{S}_i^{n+1} - \bar{S}_i^n \right\|_2}{\left\| \bar{S}_i^{n+1} \right\|_2}$
Si $Err < \epsilon$ on sort de la boucle
**fin pour**
Stockage de la source normalisée : $\bar{S}_i = \bar{S}_i^N$ avec $N$ l'indice de sortie des itérations externes.
**fin pour**

### 5.1.2 Calcul des matrices

La deuxième étape de la méthode CMS est la détermination des matrices du système linéaire. Notons que de nouvelles matrices apparaissent lorsque l'on considère plusieurs groupes d'énergie, liées au terme de transfert de l'équation de diffusion multigroupe (1.9). Aux matrices $A, B, T_a, T_f$ dont les termes sont donnés par (4.9), s'ajoutent donc les matrices de transfert $T_s$.

L'organisation de toutes les matrices se fait de manière hiérarchique, sur le modèle du solveur MINOS (classes génériques décrites section 2.3.1) : prise en compte des groupes d'énergie (indice g), puis des directions (indice d) pour les matrices faisant intervenir le courant. Or la méthode CMS multigroupe utilise des bases de flux distinctes selon le groupe d'énergie considéré, et des bases de courant qui dépendent du groupe d'énergie et de la direction. On a donc un ensemble de matrices $\left(A_d^g, B_d^g\right)_{1\leq d\leq N_d}^{1\leq g\leq N_g}$ pour le courant et $(T_a^g)^{1\leq g\leq N_g}$ pour le flux. La situation pour les matrices de fission $T_f$ et de transfert $T_s$ est particulière, puisqu'elles font intervenir un groupe de départ $g'$ et un groupe d'arrivée $g$. Les matrices de fission et de transfert sont donc indicées par g et g' : $\left(T_f^{g\to g'}, T_s^{g\to g'}\right)^{1\leq g,g'\leq N_g}$.

Les propriétés des matrices de MINOS présentées dans le chapitre 2 ne se retrouvent pas dans les matrices correspondantes de la méthode CMS : notamment les matrices $B_d$ ne sont pas composées de deux diagonales de $+1$ et $-1$, et les matrices $T_a^g$ ne sont pas diagonales.

Nous avons vu que chaque élément d'une matrice correspond à une intégrale sur $R^k \cap R^l$ de deux fonctions de base multipliées par une section efficace (voir l'équation (4.16)). Le calcul de l'intégrale est réalisé en parcourant tous les nœuds du maillage de flux ou de courant inclus dans $R^k \cap R^l$. Pour chaque noeud la contribution à l'intégrale est calculée. Même si les matrices sont creuses, leur calcul peut s'avérer coûteux, surtout quand le nombre de groupes est grand.

### 5.1.3 Résolution du système linéaire

La dernière étape de la méthode CMS est la résolution du système linéaire (4.15). Comme nous l'avons déjà évoqué, ce système a la même forme que celui résolu par MINOS (équation (2.40)), mais sa dimension est beaucoup plus petite. La technique de résolution employée est sensiblement la même. Pour la résolution du problème à valeur propre, l'algorithme itératif des puissances est utilisé. La résolution du problème à source à chaque itération et pour chaque groupe ne passe pas par l'élimination des inconnues de flux, contrairement à MINOS. En effet, cette élimination est peu coûteuse dans le cadre de MINOS car la matrice $T_a$ est diagonale donc triviale à inverser, ce qui n'est pas le cas pour la méthode CMS. Une méthode $LDL^T$ permet de résoudre le système complet (2.40) portant sur l'ensemble des inconnues de flux et de courant. La matrice associée est symétrique, mais elle n'est pas définie positive. Comme nous l'avons vu à la section 4.2, elle est creuse car tous les termes des blocs correspondant à des sous-domaines disjoints sont nuls.

Afin de disposer du flux et du courant sur le domaine global, il est nécessaire de les reconstruire à partir des vecteurs de flux et de courant solutions du système linéaire. Or nous avons vu que ces vecteurs correspondent aux coefficients des combinaisons linéaires (4.14) définissant le flux et le courant dans les espaces $V_\delta$ et $W_\delta$. En sommant les contributions des différentes fonctions de base, on obtient donc $\varphi_\delta$ et $p_\delta$. La solution globale est stockée au format de MINOS (classe Mi_Flux, voir section 2.3.1), ce qui permet de la comparer facilement à un calcul global et d'utiliser des fonctions travaillant sur un objet Mi_Flux : calcul de la distribution de puissance, calcul du flux intégré etc...

## 5.2 Application de la méthode CMS à des géométries complexes en 2D et 3D

Nous présentons maintenant l'application de la méthode CMS à la géométrie du cœur du REP en 2D et en 3D. Pour chaque résultat présenté, nous comparons la méthode CMS à un calcul direct MINOS. Les écarts sont mesurés sur le $k_{eff}$ et la puissance, définie section 2.3.2. La mesure de l'écart sur la puissance offre l'avantage de sommer des termes faisant intervenir les écarts sur les flux de tous les groupes d'énergie. On évite ainsi une présentation des écarts pour chaque groupe. L'inconvénient est que la puissance fait intervenir la section de fission, qui s'annule dans les régions non combustibles. Les écarts ne sont donc pas mesurés sur ces régions.

### 5.2.1 Test sur la géométrie 2D du REP

Comme on l'a expliqué dans la section 2.3.2, la géométrie du Réacteur à Eau sous Pression est cartésienne, car les assemblages et les cellules sont carrés. Le maillage de calcul adopté est quatre fois plus fin que le maillage physique : dans chaque direction on utilise deux mailles par cellule. Le nombre total de mailles de calcul est donc 334 084. Deux groupes d'énergie et l'élément $RT_0$ sont utilisés. Les conditions aux bords sont des conditions de flux nul.

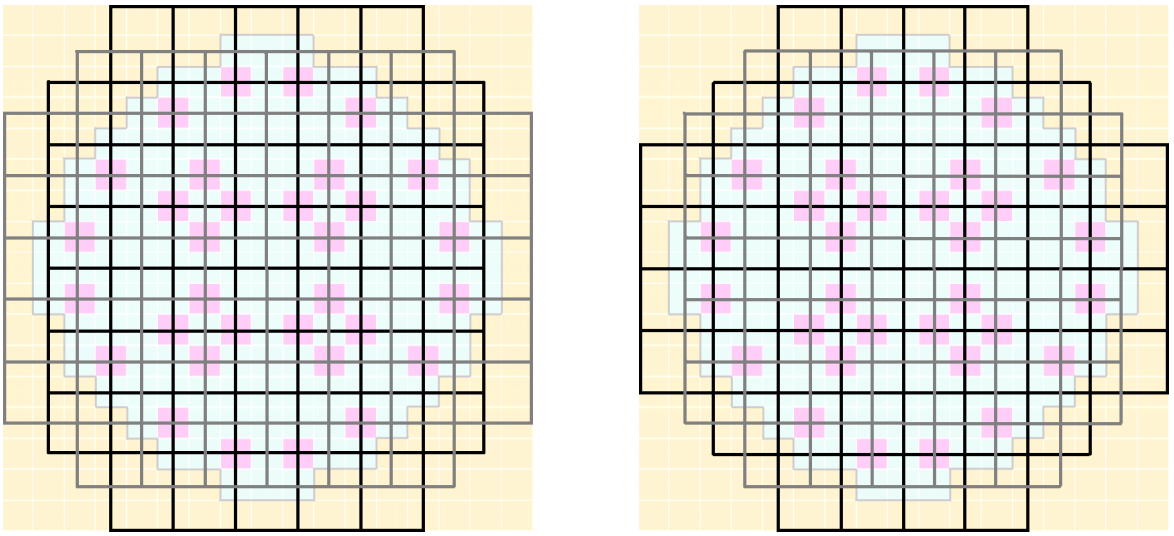
La décomposition de domaine que nous présentons sur la figure 5.1 met en œuvre 201 sous-domaines. Leurs frontières sont choisies au milieu des assemblages, pour les raisons évoquées lors des tests 1D (voir section 4.4.2). La plupart des sous-domaines ont une taille de deux assemblages dans chaque direction, sauf ceux qui sont aux bords du domaine. Cela permet d'avoir une bonne répartition de la charge de calcul sur chaque sous-domaine. Le maillage des sous-domaines est le même que celui du cœur .

On remarque que les coins du cœur ne sont pas inclus dans la décomposition de domaine. Deux raisons expliquent ce choix. La première est qu'on ne peut pas avoir un sous-domaine uniquement constitué de réflecteur. En effet la section de fission y est nul, et le problème local n'a plus de sens puisque le second membre devient nul. La deuxième raison est que cette partie du domaine est un ajout artificiel pour avoir une géométrie cartésienne. Or on constate sur la figure 2.7 que les flux rapide et thermique sont quasiment nuls sur ces quatre zones de réflecteur. Nous ne les prenons donc pas en compte dans notre calcul CMS, ce qui diminue sensiblement le nombre d'inconnues.

Afin de valider la qualité de l'approximation de la méthode CMS, nous la comparons à un calcul direct MINOS sur la même géométrie du cœur et le même maillage. Les écarts sur le $k_{eff}$ et la puissance sont mesurés. Le calcul MINOS, l'ensemble des calculs locaux et la résolution globale par la méthode CMS sont réalisés avec un grand nombre d'itérations, afin d'avoir des calculs convergés. De cette manière les écarts mesurés proviennent des écarts entre le méthode CMS et MINOS, et ne sont pas dûs à des défauts de convergence.

Pour la méthode CMS, nous présentons deux calculs avec un nombre de modes différents. Le premier utilise 4 modes de flux et 6 modes de courant sur chaque sous-domaine, alors que le deuxième est réalisé avec 9 modes de flux et 11 modes de courant sur chaque sous-domaine. L'utilisation d'un nombre de modes de courant supérieur au nombre de modes de flux est expliquée section 4.2.1. Le tableau 5.1 présente les écarts relatifs sur le $k_{eff}$ et sur la puissance. L'écart sur le $k_{eff}$ est mesuré en pcm (pour cent mille, égal à $10^{-5}$). Il est très faible dans les deux cas, inférieur à 5pcm. L'écart maximum sur la puissance est de l'ordre de 2% ($2 \times 10^{-2}$) avec 6 modes de flux, et se réduit à moins de 0,4% avec 9 modes de flux.

Les figures 5.2a et 5.2b sont des représentations graphiques de la distribution de l'écart de



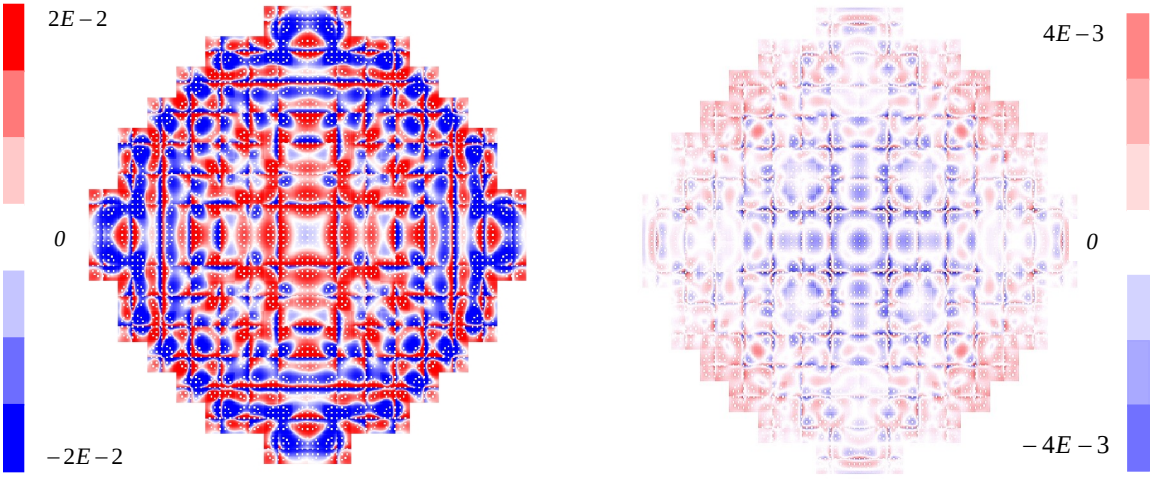
(a) Première partie des sous-domaines... (b) et les autres

FIG. 5.1: Décomposition de domaine du cœur d'un REP en 201 sous-domaines.

TAB. 5.1: Ecarts relatifs sur la puissance et sur le $k_{eff}$ entre la méthode CMS et le solveur MINOS pour le REP 2D. $k_{eff} = 1,17961$.

| | 4 modes de flux | 9 modes de flux |
|---|---|---|
| $\Delta k_{eff}$ (pcm) | 3,7 | 1,2 |
| $\|\Delta P\|_2$ | $4,3 \times 10^{-3}$ | $5,8 \times 10^{-4}$ |
| $\|\Delta P\|_\infty$ | $2,1 \times 10^{-2}$ | $3,9 \times 10^{-3}$ |

puissance. Cette écart est calculé sur chaque maille de la géométrie. La première remarque est que l'écart est plus faible dans le cas où on a plus de modes : l'approximation est meilleure car les espaces $V_\delta$ et $W_\delta$ décrits par l'équation (4.12) sont de dimensions plus grandes. La deuxième remarque est que l'on distingue dans les deux cas des segments horizontaux et verticaux, qui correspondent aux bords des sous-domaines. Cela s'explique par le fait que les fonctions de base de flux qui engendrent l'espace $V_\delta \subset L^2(R)$ sont discontinues. En effet ce sont des solutions locales obtenues avec des conditions de courant nul aux bords. Le prolongement par zéro induit donc des discontinuités aux bords des sous-domaines. On retrouve ces discontinuités dans la distribution de l'écart de puissance, puisque la puissance dépend du flux et que le flux est une combinaison linéaire des fonctions de base.

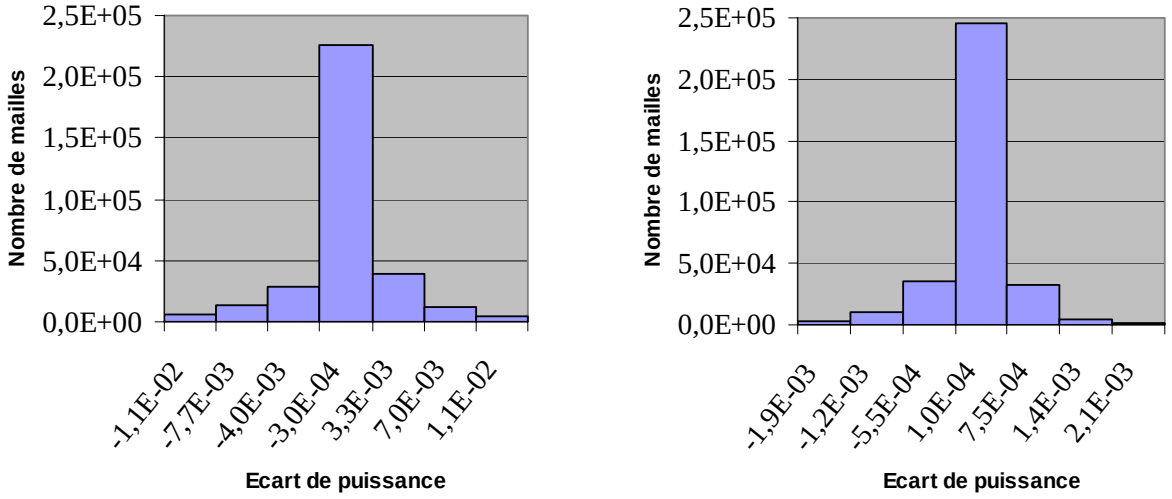



(a) 4 modes de flux, 6 modes de courant

(b) 9 modes de flux, 11 modes de courant

FIG. 5.2: Représentations graphiques des distributions de l'écart de puissance entre la méthode CMS et MINOS pour le REP 2D.

Les histogrammes 5.3a et 5.3b représentent le nombre de mailles en fonction de l'écart de puissance associé. Dans le cas où 4 modes de flux sont utilisés, l'écart est inférieur à 1% pour 95% des mailles. Avec 9 modes de flux, 95% des mailles ont un écart inférieur à 0,1%. On voit donc que l'écart est faible pour la grande majorité des mailles, et seule une petite partie d'entre elles se rapproche de l'écart maximum donné par la norme infinie.




(a) 4 modes de flux, 6 modes de courant

(b) 9 modes de flux, 11 modes de courant

FIG. 5.3: Histogrammes représentant le nombre de mailles en fonction de l'écart de puissance entre la méthode CMS et MINOS pour le REP 2D.

### 5.2.2 REP en 3D

Le domaine 3D est construit comme un ensemble de 20 plans 2D auxquels on attribue une hauteur, qui est la taille de maille dans la direction $z$. Le premier plan et le dernier sont du réflecteur, les autres ont la même géométrie que celle du REP 2D décrit à la section 2.3.2. Le maillage de calcul pour chaque plan est le même que pour le calcul 2D décrit à la section 5.2.1. Le nombre de mailles est donc vingt fois plus élevé qu'en 2D. La décomposition de domaine est la même qu'en 2D pour les axes $x$ et $y$, et nous ne décomposons pas le domaine dans l'axe des $z$. Le nombre de sous-domaines est donc toujours de 201, chaque sous-domaine ayant la même hauteur que le cœur complet. Nous présentons un calcul de diffusion à deux groupes d'énergie avec l'élément $RT_0$, et des conditions de courant nul aux bords du cœur .

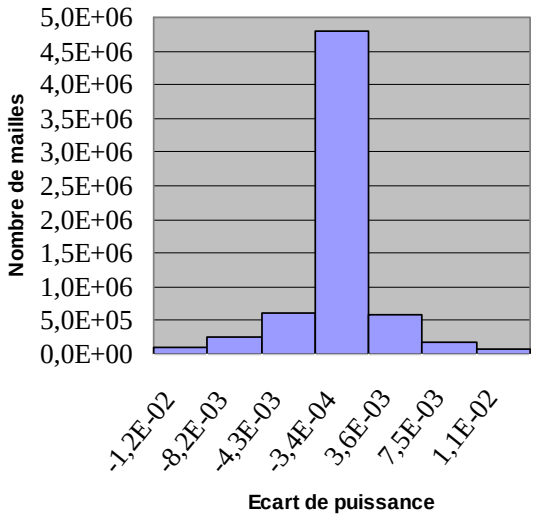
Deux tests sont présentés : le premier avec 4 modes de flux et 6 modes de courant sur chaque sous-domaine, le second avec 8 modes de flux et 10 de courant. Le tableau 5.2 fournit les écarts relatifs avec un calcul MINOS, pour le $k_{eff}$ et la puissance. On constate que ces écarts sont proches de ceux mesurés en 2D. Les histogrammes 5.4a et 5.4b fournissent la répartition du nombre de mailles selon l'écart de puissance avec MINOS. Les interprétations de ces figures sont les mêmes qu'en 2D : les écarts sont proches de ceux mesurés en 2D, et l'augmentation du nombre de modes entraîne une diminution de ces écarts. Le nombre de mailles où l'écart est proche de l'écart maximum est très faible : 95% des cellules ont un écart plus petit que 1% avec 4 modes de flux, tandis que 90% des cellules ont un écart inferieur à 0,1% avec 8 modes de flux.

### 5.2.3 L'ordre des modes

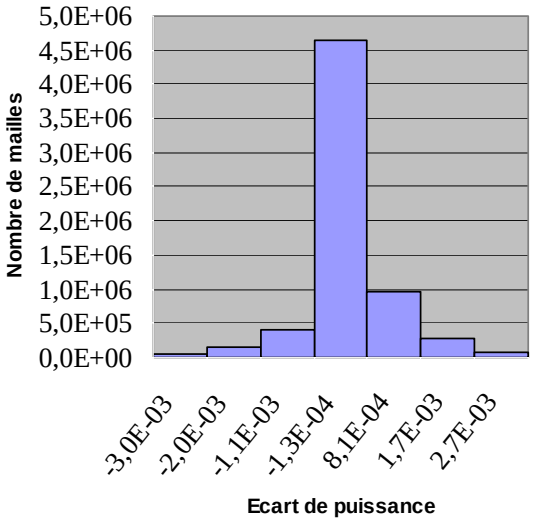
On aimerait se faire une idée des modes propres utilisés par la méthode CMS sur chaque sous-domaine. Il faut donc connaître leur ordre en fonction de leurs valeurs propres associées, puisque on ne sélectionne que les premiers. Afin de simplifier le problème, on considère en première approximation que les sections sont constantes sur tout un sous-domaine. Supposons de plus que ce sous-domaine est (en 3D) un parallélépipède de dimensions $(h_x, h_y, h_z)$. Le

TAB. 5.2: Écarts relatifs sur la puissance et sur le $k_{eff}$ entre la méthode CMS et le solveur MINOS pour le REP 3D. $k_{eff} = 1,01716$.

| | 4 modes de flux | 8 modes de flux |
|---|---|---|
| $\Delta k_{eff}$ (pcm) | 7,2 | 2,5 |
| $\|\Delta P\|_2$ | $4,9 \times 10^{-3}$ | $1,1 \times 10^{-3}$ |
| $\|\Delta P\|_\infty$ | $2,0 \times 10^{-2}$ | $3,9 \times 10^{-3}$ |




(a) 4 modes de flux, 6 modes de courant




(b) 8 modes de flux, 10 modes de courant

FIG. 5.4: Histogrammes représentant le nombre de mailles en fonction de l'écart de puissance entre la méthode CMS et MINOS pour le REP 3D.

problème de la diffusion monogroupe (2.3) est alors équivalent au problème à valeur propre (3.6) associé à l'opérateur Laplacien. Sur un domaine cartésien, les solutions sont des produits de sinus dans chaque direction :

$$\begin{aligned} u_{i_x,i_y,i_z}(x,y,z) &= sin\left(\frac{k_{i_x}\pi x}{h_x}\right) \times sin\left(\frac{k_{i_y}\pi y}{h_y}\right) \times sin\left(\frac{k_{i_z}\pi z}{h_z}\right), \\ \lambda_{i_x,i_y,i_z} &= \frac{k_{i_x}^2\pi^2}{h_x^2} + \frac{k_{i_y}^2\pi^2}{h_y^2} + \frac{k_{i_z}^2\pi^2}{h_z^2}, \quad i_x, i_y, i_z \in \mathbb{N}. \end{aligned} \tag{5.1}$$

Ces sinus deviennent des cosinus avec des conditions de courant nul aux bords. Il apparaît que l'ordre des valeurs propres dépend des dimensions du sous-domaine. Par exemple supposons que $h_x = h_y = 1, h_z = 2$. Si on classe les valeurs propres par ordre croissant, on obtient : $\lambda_1 = \lambda_{0,0,1}, \lambda_2 = \lambda_{1,0,0} = \lambda_{0,1,0} = \lambda_{0,0,2}$. Si on sélectionne les quatre premiers modes, on utilise donc trois valeurs de $k_z$ différentes, alors que l'on en utilise que deux pour $k_x$ et $k_y$.

Si on revient à des cas réalistes, on voit donc que si la hauteur du sous-domaine est beaucoup plus grande que sa largeur, les premiers modes calculés correspondront à des variations dans l'axe des $z$. Or la variation du flux selon l'axe des $z$ est la plupart du temps moins forte que les variations sur les différents plans, où les hétérogénéités sont plus grandes. On aurait donc préféré obtenir des modes variant dans les directions $x$ et $y$ plutôt que dans l'axe des $z$.

## 5.3 Parallélisation de la méthode CMS

Contrairement à beaucoup de méthodes de décomposition de domaine, la méthode de synthèse modale n'est pas une méthode itérative. Comme nous l'avons vu, elle est divisée en trois étapes : la résolution des problèmes locaux de diffusion sur tous les sous-domaines en premier lieu, le calcul des matrices en deuxième lieu et la résolution du système linéaire en troisième lieu.

La première étape est naturellement parallélisable. En effet les problèmes locaux sont entièrement indépendants. Ils sont découplés, et les conditions aux bords internes des sous-domaines sont des conditions de courant nul, non liées aux sous-domaines voisins. Si l'on a plusieurs processeurs à disposition, il suffit donc de répartir les sous-domaines, attribuant un ou plusieurs d'entre eux à chaque processeur. Aucune communication n'est nécessaire pour cette étape. L'attributions des sous-domaines aux processeurs doit se faire de façon à assurer une répartition équitable des calculs.

La deuxième étape est le calcul des matrices. Comme nous l'avons expliqué à la section 5.1.2, ces matrices sont assez nombreuses, d'autant plus que les nombres de groupes et de directions sont élevés. La première idée que nous avons eue pour la parallélisation de cette étape a été de répartir les matrices sur les processeurs disponibles. Mais cette façon de procéder est une parallélisation des tâches, alors que la répartition des problèmes locaux correspond à une parallélisation des données. Elle aboutit à une mauvaise efficacité pour deux raisons. La première est qu'elle entraîne des échanges de données importants, car tous les processeurs doivent avoir à disposition l'ensemble des flux et des courants calculés sur tous les sous-domaines. La deuxième raison est que la répartition des tâches est mauvaise, car le coût de calcul des matrices est variable. L'idée que nous avons adoptée prolonge la parallélisation des données faite par la répartition des sous-domaines : tous les processeurs participent au calcul de toutes les matrices. Chaque processeur calcule pour toutes les matrices les lignes correspondant aux sous-domaines qui lui ont été attribués. Cette technique de parallélisation permet de limiter les communications. En effet les termes des matrices sont des intégrales sur les recouvrements des sous-domaines (voir l'équation (4.16)). Les communications n'ont donc

lieu qu'entre processeurs chargés de sous-domaines qui se recouvrent. De plus, les échanges entre processeurs concernent l'ensemble des flux et des courants, mais seulement sur la zone de recouvrement.

La dernière étape de la méthode CMS, la résolution du système linéaire, est très rapide puisque la dimension du système est petite (égale au nombre total de modes calculés sur tous les sous-domaines). Elle est effectuée séquentiellement par un seul processeur. Ce processeur doit reconstituer toutes les matrices du système global. Il faut donc que les autres processeurs lui envoient les lignes des matrices qu'ils ont calculées. Même si cette étape de communication concerne l'ensemble des processeurs (*MPI_ Reduce*), elle n'est pas pénalisante car les matrices sont de petites tailles.

La figure 5.5 représente pour deux processeurs ces trois étapes du calcul parallèle. Dans notre exemple chacun des processeurs n'est associé qu'à un seul sous-domaine, mais le fait d'en avoir plusieurs ne change pas la manière de paralléliser.

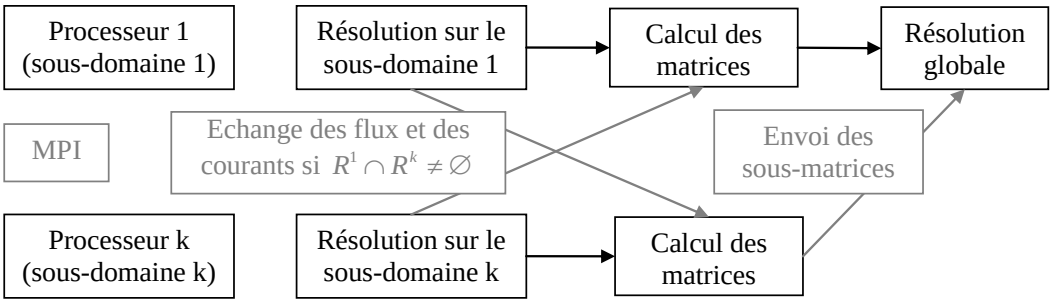



FIG. 5.5: Parallélisation de la méthode CMS.

Même si la méthode CMS présente l'avantage d'être naturellement parallélisable avec peu de communications, son coût algorithmique est nettement plus élevé que celui du solveur MINOS, et ce pour plusieurs raisons. La première est que le nombre d'opérations effectuées par MINOS est quasiment linéaire en fonction du nombre d'inconnues. La décomposition de domaine en elle-même n'apporte donc qu'un gain faible au niveau du coût algorithmique. La deuxième raison est que la méthode CMS exige des sous-domaines recouvrants, et que ces recouvrements correspondent en fait à une augmentation du nombre d'inconnues. La troisième raison est que l'algorithme utilisé pour calculer plusieurs modes propres a un coût proportionnel au nombre de modes recherchés. Il faut donc exécuter 5 fois plus d'opérations si on cherche 5 modes au lieu de 1. La quatrième raison est que le calcul des matrices est cher, comme nous l'avons mentionné à la section 5.1.2.

---

*Dans ce chapitre, nous avons commencé par décrire la programmation de la méthode CMS dans le cadre du projet APOLLO3/DESCARTES. Le solveur MINOS est utilisé pour les calculs locaux. L'algorithme de résolution du système linéaire a dû être modifié de façon à calculer plusieurs modes propres. Les matrices de la méthode CMS ont une forme similaire à celles de MINOS, et sont décrites de façon hiérarchique par les classes de conception interne de MINOS.*

*Les applications numériques sur les géométries 2D et 3D d'un cœur de réacteur REP montrent que le choix d'une décomposition de domaine adaptée à la géométrie permet d'obtenir une bonne approximation de la puissance et du $k_{eff}$ avec un nombre de modes peu élevé.*

*Cependant, même si la méthode CMS offre un fort potentiel en parallèle, le surcoût de la méthode par rapport à une résolution directe est important. En effet, le recouvrement des sous-domaines et le calcul de plusieurs modes sur chaque sous-domaine entraînent une augmentation importante du temps de calcul et de la mémoire occupée. C'est ce qui nous a amené à modifier la méthode CMS afin de diminuer le coût de la méthode. Ces modifications sont l'objet du chapitre suivant.*

---

# Chapitre 6

# Une variante de la méthode CMS : la méthode de synthèse modale factorisée

---

*Le chapitre précédent a permis de mettre en évidence les qualités d'approximation de la méthode CMS sur des géométries complexes 2D et 3D. Mais un inconvénient de la méthode est qu'elle est nettement plus chère en temps de calcul et en mémoire qu'un calcul direct MINOS. Nous proposons dans ce chapitre une modification de la méthode CMS, inspirée d'un principe de factorisation du flux en milieu périodique. Au lieu de calculer plusieurs modes sur chaque sous-domaine, seul le mode fondamental est déterminé avec des conditions de courant nul aux bords (sauf sur les bords du domaine). Les modes supplémentaires de flux sont remplacés par le produit du mode fondamental par des fonctions lisses (sinus ou cosinus), tandis que les modes de courant d'ordres plus grand que un sont remplacés par la dérivée des fonctions lisses. Les fonctions ainsi obtenues servent de fonctions de base pour les espaces d'approximation de flux et de courant.*

*Des tests sur une géométrie 1D hétérogène sont effectués afin de valider cette approche sur des domaines non périodiques où le principe de factorisation n'est pas valable. Puis des applications numériques sur des géométries complexes 2D et 3D sont présentées, afin d'évaluer la qualité d'approximation de la méthode, et de la comparer à celle de la méthode CMS. Enfin l'efficacité de la méthode en parallèle est mesurée à partir des temps de calcul sur plusieurs processeurs.*

---

Intuitivement, il semble que le fait de calculer plusieurs modes sur chaque sous-domaine nous fournit un excès d'information, sachant que l'on cherche simplement le mode fondamental sur le domaine complet. En effet, les oscillations au niveau des crayons sont bien prises en compte par le premier mode, et les modes supplémentaires donnent l'impression de ne correspondre qu'à une modification de la forme globale de la solution. Les neutroniciens ont d'ailleurs constatés que pour un cœur qui ne comporte qu'un seul type d'assemblage, le flux s'approche par le produit de deux fonctions $u$ et $\psi$. $\psi$ est la solution fondamentale du problème de diffusion en milieu infini. Elle est périodique, de période égale à un assemblage. Cette hypothèse de milieu infini est équivalente à une condition de courant nul aux bords lorsque les assemblages présentent des symétries par rapport à leurs plans médiateurs. $u$ est la solution d'un problème de diffusion homogénéisé sur le cœur : les sections efficaces sont remplacées par des coefficients homogènes constants. La fonction périodique $\psi$ permet de décrire l'allure locale du flux, tandis que $u$ est une fonction de forme donnant les variations globales.

Cette constatation est montrée rigoureusement par un théorème d'homogénéisation, dont on pourra trouver la démonstration dans [Allaire et Capdeboscq, 2000]. On suppose que le cœur est périodique, c'est à dire constitué du même assemblage qui se répète. On introduit un paramètre $\epsilon$ égal au ratio entre la taille d'un assemblage et la taille du cœur . Le i-ème mode de flux solution de l'équation de diffusion (2.1) tend, lorsque $\epsilon$ tend vers 0, vers le produit de la solution périodique fondamentale en milieu infini $\psi$ par le i-ème mode solution d'un problème de diffusion homogénéisé :

$$\varphi_i \underset{\epsilon \to 0}{\longrightarrow} u_i \times \psi. \tag{6.1}$$

$u_i$ est la solution du problème homogénéisé suivant :

$$\begin{cases} -\overrightarrow{\nabla}.\left(\bar{D}\overrightarrow{\nabla} u_i\right) = \dfrac{1}{\bar{\lambda}_i}\bar{\sigma} u_i \quad \text{sur } R, \\ u_i = 0 \quad \text{sur } \partial R. \end{cases} \tag{6.2}$$

$\bar{\sigma}$ est un coefficient constant et $\bar{D}$ est une matrice de coefficients, de dimension égale au nombre de directions. On trouvera l'expression de ces coefficients homogénéisés dans [Allaire et Capdeboscq, 2000]. Notons que ce théorème de factorisation a un analogue pour l'équation du transport, démontré dans [Allaire et Bal, 1999]. La généralisation à plusieurs groupes d'énergie présente une propriété très intéressante : la fonction de forme est solution d'un problème de diffusion à coefficients constants, et à un seul groupe.

Mais cette technique d'homogénéisation n'est valable que sur des cœurs périodiques constitués d'assemblages équivalents, ce qui est rarement le cas. Pour des cœurs non périodiques, des facteurs de discontinuité sont souvent utilisés, afin de prendre en compte la différence de niveau entre les flux de part et d'autre d'une interface entre deux assemblages différents. Une approche avec des facteurs de discontinuité sur le courant est proposé dans [Siess, 2004]. Les applications numériques sont encourageantes, car l'allure globale du flux est bien représentée. Mais le modèle d'homogénéisation atteint ses limites pour des cas réalistes où l'on s'éloigne de l'hypothèse de périodicité.

## 6.1 Définition de nouvelles fonctions de base

Notre objectif est de s'inspirer de cette factorisation des modes propres solutions de l'équation de diffusion, afin d'éviter le calcul des modes locaux d'ordres plus grand que un dans la méthode CMS. L'idée est d'obtenir de nouvelles fonctions de base ne nécessitant que le calcul du mode fondamental sur chaque sous-domaine. Etant donné que l'on se place sur des

sous-domaines non périodiques, et de petite taille ($\epsilon$ est dans ce cas grand), le but n'est pas d'obtenir une bonne approximation des modes d'ordres supérieurs. On souhaite simplement avoir des fonctions de base qui prennent bien en compte les variations locales du flux.

On définit donc sur chaque sous-domaine $R^k$ des fonctions qui s'expriment comme le produit du flux en milieu infini, et de solutions d'un problème homogène de diffusion à coefficients constants. Si on suppose $\bar{D}, \bar{\sigma}$ constants sur $R^k$, l'opérateur du Laplacien se substitue à l'opérateur de diffusion dans (6.2). On considère sur $R^k$ le problème suivant :

$$\begin{cases} -\bar{D}\Delta u_i^k = \dfrac{1}{\overline{\lambda}_i^k}\bar{\sigma}u_i^k \quad \text{sur } R^k, \\ \dfrac{\partial u_i^k}{\partial n} = 0 \quad \text{sur } \partial R^k \setminus \partial R, \\ u_i^k = 0 \quad \text{sur } \partial R^k \cap \partial R. \end{cases} \qquad 2 \leq i \leq N^k \tag{6.3}$$

Ce problème peut être résolu analytiquement : les solutions sont des sinus ou des cosinus. Par ailleurs, on note $(p^k, \varphi^k)$ la solution fondamentale sur $R^k$ du problème de diffusion non homogénéisé (4.10). On considère alors les fonctions définies de la manière suivante :

$$\begin{cases} \tilde{\varphi}_1^k = \varphi^k \quad \text{sur } R^k \\ \tilde{\varphi}_1^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \quad \begin{cases} \tilde{\varphi}_i^k = \varphi^k \times u_i^k \quad \text{sur } R^k \\ \tilde{\varphi}_i^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \qquad 1 \leq k \leq K, 2 \leq i \leq N_\varphi^k. \tag{6.4}$$

Ces fonctions sont les fonctions de base de flux utilisées pour la méthode FCMS (*Factorized CMS method*). On remarque que la première fonction de base sur chaque sous-domaine est la même que pour la méthode CMS : c'est la solution fondamentale du problème local en milieu infini.

Afin de se limiter au calcul du mode fondamental, il est nécessaire de modifier également les fonctions de base de courant. Là-aussi, on conserve la première fonction de base de courant sur chaque sous-domaine, égale au mode fondamentale de courant en milieu infini prolongé par zéro. Mais le courant ne peut pas se factoriser de la même manière que le flux. Pour remplacer les solutions d'ordres supérieurs, on utilise le fait que le courant est lié à la dérivée du flux par la loi de Fick (équation (1.6)) : $\overrightarrow{p} = -D\overrightarrow{\nabla}\varphi$. Les fonctions que l'on va utiliser sont simplement la dérivée des solutions de (6.3), c'est à dire des sinus et cosinus que l'on prolonge par zéro pour être définis sur $R$. Les fonctions de base de courant que l'on utilise sont donc :

$$\begin{cases} \tilde{p}_{1,d}^k = p_d^k \quad \text{sur } R^k \\ \tilde{p}_{1,d}^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \quad \begin{cases} \tilde{p}_{i,d}^k = \dfrac{\partial u_i^k}{\partial d} \quad \text{sur } R^k \\ \tilde{p}_{i,d}^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \qquad 1 \leq k \leq K, 2 \leq i \leq N_p^k. \tag{6.5}$$

On remarque que les variations locales du courant ne sont données que par la première fonction de base. Toutes les autres sont des fonctions de forme qui permettent de déterminer l'allure globale du courant.

Ces fonctions de base de flux et de courant engendrent les espaces d'approximation $V_\delta$ et $W_\delta$ définis par (4.12). Le changement des fonctions de base de ces espaces est la seule modification apportée par la méthode FCMS par rapport à la méthode CMS. Les étapes suivantes de la méthode sont celles décrites dans la section 4.2.

Notons que, quelque soit le nombre de fonctions de base de flux et de courant que l'on souhaite utiliser, seul le mode fondamental doit être calculé sur chaque sous-domaine. Le coût

des calculs locaux ne dépend donc pas de la dimension des espaces $V_\delta$ et $W_\delta$, contrairement à la méthode CMS. Ce n'est pas le cas des matrices dont le calcul est d'autant plus coûteux que le nombre de fonctions de base est grand.

## 6.2 Applications numériques

### 6.2.1 Applications numériques en 1D

Les applications numériques que nous allons présenter vont nous permettre de voir si la méthode FCMS fournit une bonne approximation de la solution du problème de la diffusion critique. En particulier, il faut vérifier si l'usage d'un seul mode propre par sous-domaine, et le remplacement des modes d'ordres supérieurs par des fonctions de forme ne pénalisent pas trop la qualité d'approximation par rapport à la méthode CMS. Les premiers tests que nous présentons utilisent la géométrie 1D décrite à la section 4.4.1, et un seul groupe d'énergie est employé.

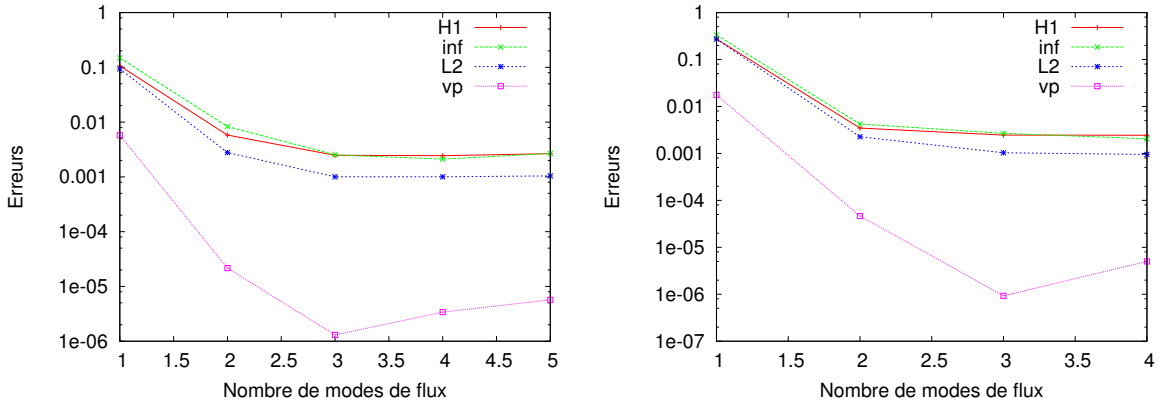
Nous choisissons la décomposition de domaine qui nous a semblé être la plus efficace pour la méthode CMS. Cette décomposition, présentée sur la figure 4.7, utilise 17 sous-domaines qui se recouvrent totalement. Nous utilisons dans un premier temps les fonctions de base décrites à l'équation (6.4) pour le flux et à l'équation (6.5) pour le courant. Les fonctions de forme $u_i^k$ sont des sinus ou des cosinus, selon les conditions aux bords imposées sur le sous-domaine $R^k$. Dans le cadre de la méthode FCMS, nous continuons à nommer par "modes" ces fonctions de base, bien qu'elles ne soient pas solution d'un problème à valeur propre.

Les premiers tests que nous présentons utilisent sur chaque sous-domaine un mode de courant supplémentaire par rapport au nombre de modes de flux : $N_p^k = N_\varphi^k + 1, \forall k$. La figure 6.1a présente les erreurs relatives sur le $k_{eff}$ et sur le flux en norme $H^1$, $L^2$ et $L^\infty$ par rapport à la solution de référence. Il apparaît que pour un petit nombre de modes, l'approximation du mode fondamentale est très correcte. Par exemple, avec deux modes de flux et trois modes de courant sur chaque sous-domaine, la différence des flux en norme infinie est de 0,8% tandis que la différence des $k_{eff}$ est de l'ordre de 2pcm. Cependant, lorsque l'on compare ces résultats à ceux de la méthode CMS avec la même décomposition de domaine, à même nombre de modes, la méthode CMS est plus précise. Surtout, la convergence de la méthode FCMS semble bornée lorsque l'on augmente le nombre de modes, ce qui n'est pas le cas pour la méthode CMS. Nous avons vu à la section précédente que seuls les modes fondamentaux locaux fournissent une description locale du courant dans la méthode FCMS, les autres fonctions de base de courant étant simplement des sinus ou des cosinus. Il est donc intéressant de voir si l'augmentation du nombre de modes de courant améliore l'approximation. La figure 6.1b présente les erreurs relatives entre le calcul direct de référence et la méthode FCMS avec $N_p^k = N_\varphi^k + 3, \forall k$. Même si les approximations sont globalement un peu meilleures qu'avec $N_p^k = N_\varphi^k + 1$, les deux fonctions de base de courant ajoutées n'apportent pas un grand gain en précision.

Bien qu'on ne dispose pas d'un principe de factorisation qui permette de prendre en compte les variations locales du courant dans les fonctions de base autres que les modes fondamentaux, il est en revanche possible de construire ces fonctions de base à partir de celles de flux. En effet, dans le système linéaire (2.40), on peut voir que dans toutes les directions $d$, on a :

$$P_d = A_d^{-1} B_d \Phi. \tag{6.6}$$

En fait cette relation correspond à la forme duale de la loi de Fick : $\overrightarrow{p} = -D\overrightarrow{\nabla}\varphi$. On propose donc d'adopter des nouvelles fonctions de base de courant, définies comme suit :




(a) $N_p^k = N_\varphi^k + 1, \forall k$ (b) $N_p^k = N_\varphi^k + 3, \forall k$

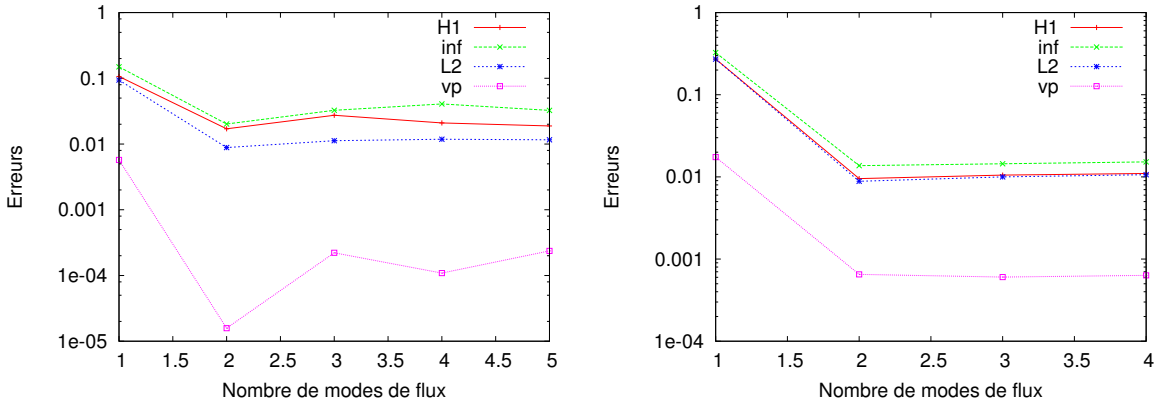
FIG. 6.1: Erreurs relatives d'approximation pour une décomposition en 17 sous-domaines, avec les fonctions de base (6.4) pour le flux et (6.5) pour le courant.

$$\begin{cases} \tilde{p}_{1,d}^k = p_d^k \quad \text{sur } R^k \\ \tilde{p}_{1,d}^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \qquad 1 \le k \le K,$$

$$\begin{cases} \tilde{p}_{i,d}^k = -D\overrightarrow{\nabla}\varphi_i^k \quad \text{sur } R^k \\ \tilde{p}_{i,d}^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \qquad 1 \le k \le K, 2 \le i \le N_\varphi^k, \tag{6.7}$$

$$\begin{cases} \tilde{p}_{i,d}^k = \dfrac{\partial u_i^k}{\partial d} \quad \text{sur } R^k \\ \tilde{p}_{i,d}^k = 0 \quad \text{sur } R \setminus R^k \end{cases} \qquad 1 \le k \le K, N_\varphi^k < i \le N_p^k.$$

La figure 6.2a présente les erreurs relatives d'approximation de la méthode FCMS avec ces fonctions de base de courant, et $N_p^k = N_\varphi^k + 1, \forall k$. L'approximation est décevante, et bien moins bonne que celle obtenue avec les fonctions de base de courant (6.5). Même avec 5 modes de flux et six modes de courant, l'erreur est de l'ordre de 3% pour la norme infinie du flux, et de 24pcm pour le $k_{eff}$. La figure 6.2b présente les erreurs relatives d'approximation, cette fois-ci avec $N_p^k = N_\varphi^k + 3, \forall k$. Les deux fonctions de base de courant supplémentaires améliorent un peu l'approximation. Mais les fonctions de base de courant (6.7) s'avèrent dans tous les cas moins efficaces que les fonctions (6.5). De plus leur détermination est coûteuse puisqu'elle demande la résolution d'un système linéaire du type (6.6), $\forall 1 \le k \le K, \forall 2 \le i \le N_\varphi^k$. Nous utiliserons donc dorénavant les fonctions de base de courant (6.5).




(a) $N_p^k = N_\varphi^k + 1, \forall k$




(b) $N_p^k = N_\varphi^k + 3, \forall k$

FIG. 6.2: Erreurs relatives d'approximation pour une décomposition en 17 sous-domaines, avec les fonctions de base (6.4) pour le flux et (6.7) pour le courant.

### 6.2.2 Applications numériques sur des géométries complexes 2D et 3D

#### Le REP en 2D

En 2 dimensions, se pose la question du choix des fonctions de forme $u_i^k$. En effet ces fonctions sont des produits de sinus ou de cosinus dans chaque direction, en fonction des conditions imposées aux bords des sous-domaines. Supposons que nous nous plaçons sur un sous-domaine interne, tel que $\partial R^k \cap \partial R = \emptyset$. Les conditions aux bords sont alors des conditions de courant nul, et l'expression des $u_i^k$ et des $p_{i,d}^k$ est :

$$\begin{cases} u_i^k(x,y) = cos\left(\dfrac{\pi i_x}{h_x}x\right) cos\left(\dfrac{\pi i_y}{h_y}y\right), \\ p_{i,x}^k(x,y) = sin\left(\dfrac{\pi(i_x+1)}{h_x}x\right) cos\left(\dfrac{\pi i_y}{h_y}y\right), \\ p_{i,y}^k(x,y) = cos\left(\dfrac{\pi i_x}{h_x}x\right) sin\left(\dfrac{\pi(i_y+1)}{h_y}y\right), \end{cases} \quad i_x, i_y \geq 0. \tag{6.8}$$

Le choix à faire est l'ordre dans lequel sont incrémentés les $i_x$ et $i_y$. Le tableau 6.1 donne les valeurs que nous utilisons pour le calcul du REP 2D. Pour les sous-domaines carrés, tels que

TAB. 6.1: Valeurs des indices $i_x$ et $i_y$ associés aux fonctions de forme d'indice i.

| i | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $i_x$ | 0 | 1 | 0 | 2 | 0 | 1 | 3 | 0 | 4 | 0 | 2 |
| $i_y$ | 0 | 0 | 1 | 0 | 2 | 1 | 0 | 3 | 0 | 4 | 2 |

$h_x = h_y$, ce choix respecte l'ordre des valeurs propres croissantes du problème de diffusion homogénéisé (6.3), dont les solutions sont les fonctions (6.8). On remarque que les valeurs propres ont une multiplicité plus grande que un. La possibilité de choisir l'ordre des fonctions de forme est un avantage de la méthode FCMS par rapport à la méthode CMS. En effet, la méthode CMS fait intervenir les modes propres locaux dans l'ordre des valeurs propres, et nous avons vu à la section 5.2.3 que cet ordre ne correspond pas toujours à l'ordre souhaité.

Le calcul du REP 2D est décrit à la section 5.2.1. La géométrie, le maillage de calcul et la décomposition de domaine sont les mêmes que ceux utilisés pour la méthode CMS. Nous avons constaté en 1D que le nombre de modes de courant doit être sensiblement supérieur au nombre de modes de flux pour avoir une bonne approximation. Nous présentons des résultats avec 6 modes de flux et 11 modes de courant sur chaque sous-domaine.

La démarche pour valider la méthode sur le cœur du REP est la même que pour la méthode CMS : on compare le $k_{eff}$ et la puissance renvoyés par un calcul FCMS, à ceux fournis par un calcul MINOS convergé. On retrouve dans la distribution de l'écart de puissance représentée sur la figure 6.3, les segments parallèles aux axes déjà constatés pour la méthode CMS, qui correspondent aux bords des sous-domaines. Quant à la répartition des mailles en fonction de l'écart de puissance, l'histogramme 6.4 a la même allure que ceux de la méthode CMS. Comme on peut le voir dans le tableau 6.2, les différences relatives de $k_{eff}$ et de puissance sont très faibles.

Afin de mesurer l'impact des modes propres fondamentaux locaux dans la précision de la méthode, nous avons ajouté à ce tableau une colonne correspondant à la suppression de ces modes. Ainsi on élimine les modes fondamentaux des bases de flux et de courant $V_\delta$ et $W_\delta$ (engendrées par les fonctions (6.4) et (6.5)), et on ne multiplie pas sur chaque sous-domaine les modes lisses de flux par le flux fondamental. On remarque que l'écart sur le $k_{eff}$ reste assez faible. Par contre l'écart relatif sur la puissance est très important puisque son maximum est de 34%. Les modes fondamentaux sont donc primordiaux pour la qualité d'approximation de la méthode FCMS.

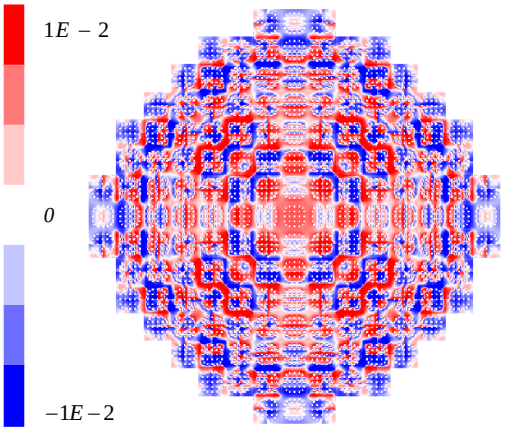



FIG. 6.3: Représentation graphique de la distribution de l'écart de puissance entre la méthode FCMS et MINOS sur le REP 2D.

TAB. 6.2: Ecarts relatifs sur la puissance et sur le $k_{eff}$ entre la méthode FCMS et le solveur MINOS pour le REP 2D. $k_{eff} = 1,17961$.

| | 6 modes de flux, 11 modes de courant | Modes lisses uniquement |
|---|---|---|
| $\Delta k_{eff}$ (pcm) | 1,9 | 52 |
| $\|\Delta P\|_2$ | $3,1 \times 10^{-3}$ | $1,2 \times 10^{-1}$ |
| $\|\Delta P\|_\infty$ | $1,0 \times 10^{-2}$ | $3,4 \times 10^{-1}$ |

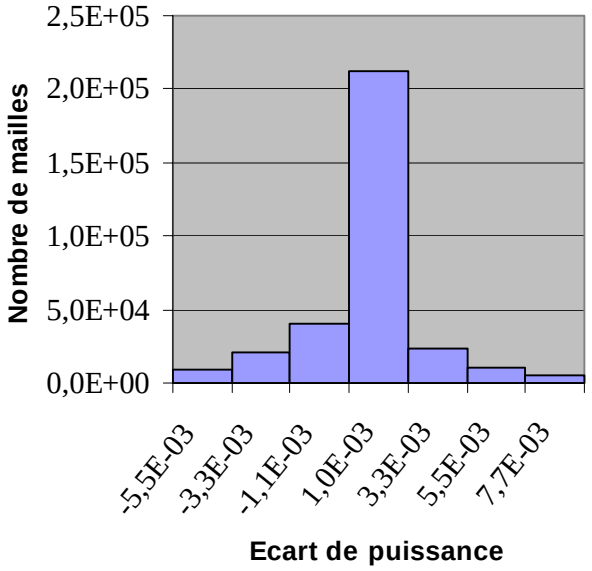



FIG. 6.4: Histogramme représentant le nombre de mailles en fonction de l'écart de puissance entre la méthode FCMS et MINOS pour le REP 2D.

### Le REP en 3D

Nous présentons maintenant une application de la méthode FCMS sur le REP 3D. Le calcul est décrit à la section 5.2.2. Les fonctions de forme sont similaires aux fonctions (6.8) en 2D, mais on ajoute un cosinus pour la variable z, et un indice associé $i_z$. La composante $p^k_{i,z}$ du courant est un produit de deux cosinus dans les directions $x$ et $y$, et d'un sinus dans la direction $z$. Étant donné que la variation du flux est faible dans l'axe des $z$, on prend toujours $i_z = 0$, ce qui revient en fait à ne pas introduire de dépendance à la variable z pour $u^k_i, p^k_{i,x}, p^k_{i,y}$. Nous utilisons le tableau 6.1 pour les valeurs des indices $i_x, i_y$.

Les écarts de puissance et de $k_{eff}$ avec un calcul MINOS convergé sont décrits par l'histogramme 6.5 et le tableau 6.3. Les interprétations sont les mêmes que celles données pour la méthode CMS : les $k_{eff}$ sont très proches et l'écart de puissance est faible, l'écart maximum étant inférieur à 1%.

TAB. 6.3: Ecarts relatifs sur la puissance et sur le $k_{eff}$ entre la méthode FCMS et le solveur MINOS pour le REP 3D. $k_{eff} = 1,01716$.

| | 6 modes de flux, 11 modes de courant |
|---|---|
| $\Delta k_{eff}$ (pcm) | 5,67 |
| $\|\Delta P\|_2$ | $3,5 \times 10^{-3}$ |
| $\|\Delta P\|_\infty$ | $9,1 \times 10^{-3}$ |

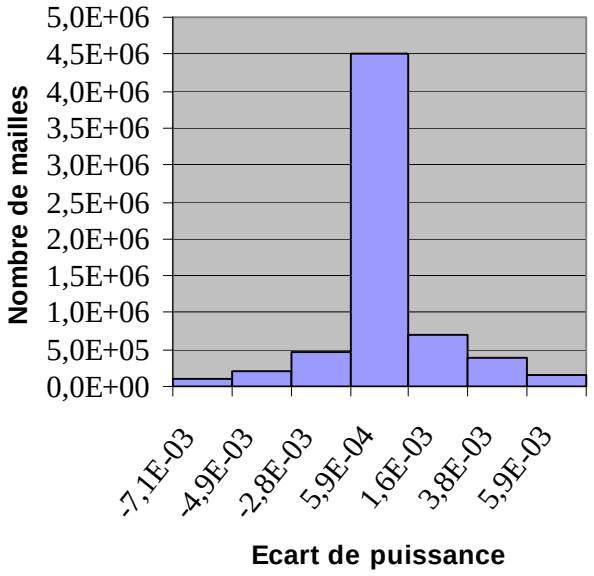



FIG. 6.5: Histogramme représentant le nombre de mailles en fonction de l'écart de puissance entre la méthode FCMS et MINOS pour le REP 3D.

### RJH en 2D

Nous appliquons maintenant la méthode FCMS au cœur 2D du RJH, décrit à la section 2.3.2. A la différence du REP, la géométrie n'est pas cartésienne et le cœur ne présente aucun axe de symétrie. Le maillage de calcul est le même que le maillage géométrique, constitué de un million de mailles carrées. Le calcul est un calcul de diffusion à six groupes d'énergie avec l'élément $RT_0$. Le nombre d'inconnues de flux est donc six fois plus grand que le nombre de mailles.

On a vu que dans le cas du REP, les sous-domaines ont été choisis de manière à ce que leurs bords soient au milieu des assemblages, où la condition de courant nul est à priori mieux vérifiée qu'à l'interface entre deux assemblages. Cette considération physique a pu être prise en compte dans le cas du REP, car sa géométrie est cartésienne. Il est en effet possible de définir des sous-domaines rectangulaires dont les bords sont au milieu des assemblages. Ce n'est pas le cas pour le RJH. Une autre contrainte est que le réflecteur occupe une grande place dans la géométrie. Il est donc nécessaire d'avoir de grands sous-domaines puisque on ne peut pas définir un sous-domaine sur une zone où la section de fission est partout nulle, ce qui est le cas du réflecteur. La décomposition de domaine que nous choisissons utilise 9 sous-domaines recouvrants de même taille, de façon à avoir une bonne répartition des calculs. Ces sous-domaines sont représentés sur la figure 6.6.

Notre premier test utilise 6 modes de flux et 11 modes de courant sur chaque sous-domaine, et les modes lisses sont déterminés à l'aide du tableau 6.1. Les écarts relatifs avec un calcul direct MINOS sont grands : 1150pcm pour le $k_{eff}$, 8% pour l'écart maximum de puissance et 3% pour l'écart de puissance en norme $L^2$. Un deuxième test, réalisé avec 11 modes de flux et 14 modes de courant, ne permet pas de diminuer ces écarts. Cette mauvaise approximation par la méthode FCMS vient notamment du saut de la distribution de puissance à l'interface des sous-domaines. Ce saut, visible sur la figure 6.7, est dû à la discontinuité des flux pour les six groupes d'énergie.

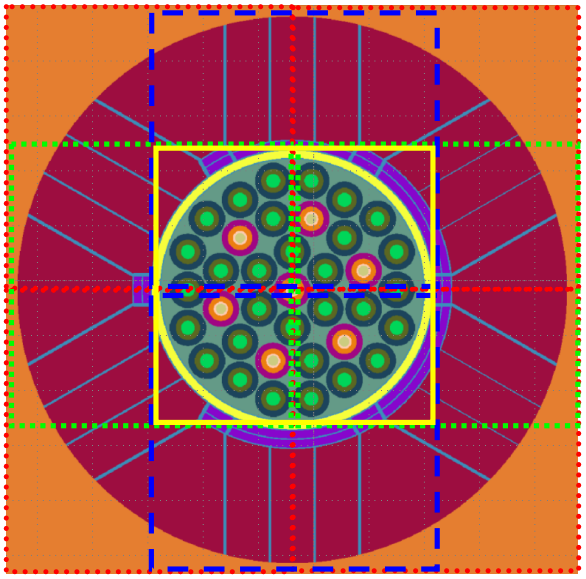


FIG. 6.6: Décomposition de domaine du RJH.

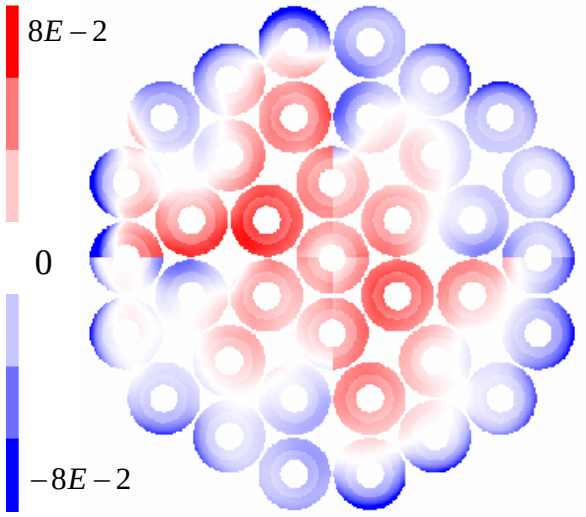



FIG. 6.7: Écart de puissance entre la méthode FCMS et MINOS sur le RJH 2D.

Une forte augmentation du nombre de modes de flux et de courant permettrait sans doute de réduire ces discontinuités. Mais il faudrait probablement utiliser un algorithme d'élimination des modes liés pour que le système reste inversible (voir test 1D section 4.4.2). De plus, le calcul des matrices deviendrait très coûteux en temps CPU. En effet nous avons vu que les matrices de disparition et de production dépendent du groupe $g$ des neutrons incidents, et du groupe $g'$ des neutrons produits, avec $1 \leq g, g' \leq N_g$. Le calcul du RJH étant à six groupes, ces matrices doivent donc être calculées 36 fois avec des sections différentes. A cela s'ajoutent les deux directions de calcul pour les matrices de disparition $A_d$, et pour les matrices $B_d$ (voir système linéaire (2.40)).

Il apparaît donc que la méthode FCMS n'est pas bien adaptée à des géométries telles que celle du RJH. L'utilisation de sous-domaines non rectangulaires et mieux adaptés à la géométrie du cœur permettrait sans doute d'obtenir de meilleurs résultats. La résolution des problèmes locaux sur des géométries non cartésiennes n'est pour l'instant pas possible avec MINOS, mais l'adaptation du solveur à des éléments triangulaires ([Baudron *et al.*, 2007]) permettra de prendre en compte une variété de géométries beaucoup plus grande.

## 6.3 Efficacité de la méthode FCMS en parallèle

Le principe de la parallélisation de la méthode FCMS est le même que pour la méthode CMS (section 5.3). Sur chaque sous-domaine, un seul mode est calculé et des modes lisses (sinus et cosinus) sont introduits via une classe *Mode_Lisse*.

Nous avons testé la méthode FCMS en parallèle sur un REP en 3D. Le maillage utilise 20 plans dans l'axe des $z$, la maille dans les plans $x, y$ correspondant à une cellule. Le cœur est décomposé en 49 sous-domaines de même taille : les axes $x$ et $y$ sont divisés chacun en 7, la hauteur des sous-domaines étant égale à celle du cœur . Sur chaque sous-domaine 11 modes de courant et 6 modes de flux sont utilisés. Le critère d'arrêt pour les calculs locaux est de $10^{-5}$ sur le $k_{eff}$ et de $10^{-4}$ sur la source de fission.

Les tests ont été réalisés sur le calculateur Tantale décrit à la section 3.1.2. L'histogramme 6.8 représente les temps de calcul en fonction du nombre de processeurs utilisé, et l'efficacité associée. Les sous-domaines sont répartis équitablement sur tous les processeurs. En référence on donne le temps de calcul de MINOS, avec un critère d'arrêt de $10^{-5}$ sur le $k_{eff}$ et de $10^{-4}$ sur la source de fission.

La première remarque est que sur un seul processeur, le temps de calcul de la méthode FCMS est beaucoup plus grand que celui de MINOS. Comme nous l'avons vu au chapitre 2, la complexité de MINOS est linéaire en fonction du nombre d'inconnues. La décomposition de domaine n'apporte donc pas de gain en terme de complexité, et les recouvrements des sous-domaines augmentent le nombre total d'inconnues. De plus, le calcul des matrices dans MINOS est très rapide, alors qu'il représente une part non négligeable du temps de calcul dans la méthode FCMS.

Mais l'augmentation du nombre de processeurs permet de diminuer les temps de calcul de la méthode FCMS, et de rejoindre MINOS avec 8 processeurs et une efficacité associée de près de 80%. L'augmentation du nombre de processeurs entraîne une diminution de l'efficacité, mais l'utilisation de 25 processeurs permet de passer en dessous de 100 secondes avec une efficacité de l'ordre de 50%. Au-dessus, l'efficacité se dégrade trop, et l'utilisation de 49 processeurs n'apporte que peu de gain, avec une efficacité de 30%.

On constate dans la courbe de l'efficacité des variations significatives. Par exemple, elle s'améliore assez nettement quand on passe de 24 à 25 processeurs. Cela s'explique par la répartition des calculs locaux. En effet, avec 24 processeurs, chacun traite deux sous-domaines,

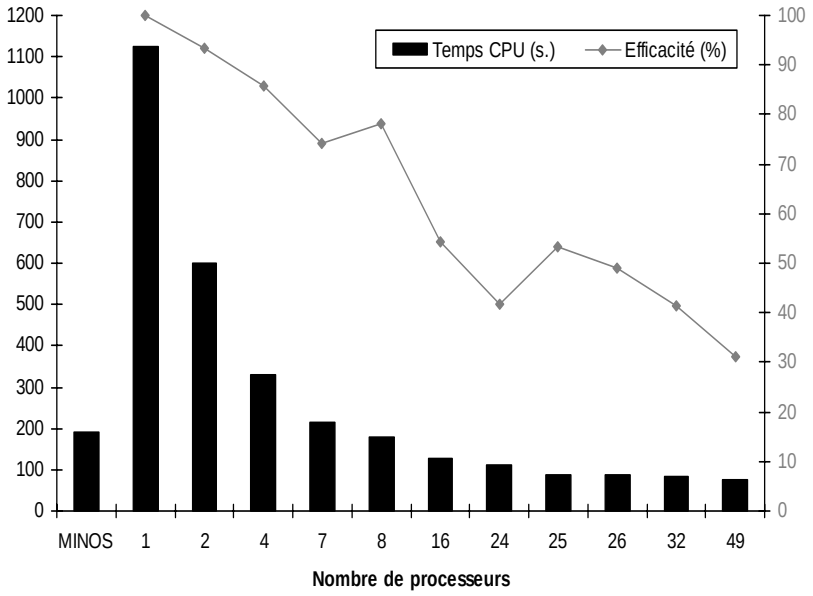



FIG. 6.8: Temps CPU et efficacité de la méthode FCMS, en fonction du nombre de processeurs.

sauf un processeur qui en a trois. Tous les processeurs doivent donc attendre la fin du calcul sur ce troisième sous-domaine avant de passer au calcul des matrices. A l'inverse, avec 25 processeurs, chacun a deux sous-domaines à traiter, sauf un qui n'en a qu'un. C'est le seul processeur en attente avant le calcul des matrices, ce qui explique l'amélioration de l'efficacité.

Les temps de calcul de la méthode FCMS dépendent fortement des paramètres choisis : recouvrement des domaines, nombre de modes, critère d'arrêt pour les calculs locaux. Nous avons choisi un compromis de manière à avoir une approximation correcte en limitant les temps de calcul et l'occupation mémoire. Le but de ces tests est d'évaluer la qualité de la parallélisation de la méthode FCMS, plutôt que de comparer les temps de calcul de la méthode FCMS au temps de MINOS. En effet, cette comparaison n'a pas réellement de sens car la précision des deux méthodes n'est pas la même : MINOS fournit une solution plus proche de la solution "exacte". Le critère d'arrêt que nous avons pris pour le calcul global MINOS et pour les calculs locaux est le même. Il est donc peu probable d'obtenir une meilleure approximation par la méthode FCMS. Pour avoir une approximation comparable, il aurait fallu relâcher le critère d'arrêt pour le calcul MINOS, ce qui aurait diminué le temps de calcul associé.

## 6.4 Perspectives pour les méthodes CMS et FCMS

Nous envisageons plusieurs perspectives d'utilisation des méthodes CMS et FCMS. La première est l'application de la méthode CMS (ou FCMS) à des calculs de cinétique. En effet, on peut imaginer utiliser les mêmes solutions locales pour former les fonctions de base de plusieurs pas de temps. Pour ces pas de temps il ne resterait donc plus qu'à construire les matrices, et à résoudre le système linéaire.

La deuxième perspective serait de constituer une "bibliothèque" de modes locaux. Certaines configurations de sous-domaines pourraient être calculées une fois pour toutes, et réutilisées dans différents calculs.

La troisième perspective est l'utilisation de techniques du type "bases réduites". L'idée serait de choisir un nombre limité de fonctions de base représentatives, et de les utiliser sur plusieurs sous-domaines. Par exemple, les mêmes fonctions de base pourraient être utilisées sur des sous-domaines ayant des géométries proches. Cette technique permettrait de réduire les temps de calcul et la mémoire allouée, pour deux raisons. La première est qu'on ne ferait des calculs locaux que sur une partie des sous-domaines. La deuxième est que certains termes des matrices seraient égaux, et ne seraient donc à calculer qu'une fois.

Par ailleurs, il serait intéressant d'appliquer ces méthodes à d'autres modèles. En effet, elles ne sont pas liées au modèle de la diffusion. L'application aux équations $SP_N$, qui correspondent à des équations de diffusion couplées, demande peu d'efforts car le code que nous avons développé contient déjà des boucles sur les harmoniques.

Les méthodes CMS et FCMS offrent également la possibilité d'utiliser des maillages différents sur les sous-domaines, alors que MINOS ne peut pas modifier localement les tailles des mailles puisqu'elles sont définies dans chaque direction de calcul. Ainsi, il est possible d'utiliser un maillage plus fin sur un sous-domaine où le flux varie fortement, ou sur un sous-domaine qui présente un intérêt particulier comme le pic de puissance.

Enfin, le couplage de différents modèles est envisageable. On peut par exemple imaginer traiter certains sous-domaines avec une approximation $SP_3$, tandis que d'autres utilisent une approximation de diffusion. La résolution globale serait faite en $SP_3$.

*Nous avons montré que les méthodes CMS et FCMS permettent de résoudre le problème de la diffusion critique multigroupe sur des géométries complexes de cœurs industriels. Le choix d'une décomposition de domaine adaptée permet d'obtenir une bonne approximation de la solution avec un nombre restreint de modes.*

*La méthode FCMS présente plusieurs atouts par rapport à la méthode CMS. Le premier est qu'elle nécessite uniquement le calcul du mode fondamental sur chaque sous-domaine : les coûts en temps de calcul et en mémoire sont donc fortement réduits par rapport à la méthode CMS. De plus, les résultats numériques montrent que pour une précision équivalente, même si le nombre de fonctions de base de courant requis par la méthode FCMS est plus grand que pour la méthode CMS, le nombre de fonctions de base de flux est par contre du même ordre pour les deux méthodes. La factorisation du flux utilisée dans la méthode FCMS est donc efficace, et les modes fondamentaux locaux sont suffisants pour décrire correctement le comportement local de la solution globale. Enfin la méthode FCMS autorise le choix des fonctions que l'on ajoute aux bases de flux et de courant, alors que la sélection des modes propres par la méthode CMS se fait selon l'ordre des valeurs propres. Les méthodes CMS et FCMS ont l'avantage de ne pas être itératives, et la résolution des problèmes locaux constitue une étape indépendante du calcul des matrices et de la résolution globale. De plus les calculs locaux sont indépendants entre eux. Il est donc possible de les réaliser séparément, de stocker les flux et les courants locaux et de les réutiliser ultérieurement pour calculer les matrices et résoudre le système linéaire global.*

*Mais ces méthodes restent chères en temps de calcul et en mémoire, et l'utilisation de plusieurs processeurs en parallèle est nécessaire pour obtenir un temps de calcul du même ordre que celui du solveur direct MINOS. L'efficacité est bonne avec peu de processeurs, mais se dégrade lorsqu'on augmente leur nombre. Certes, la parallélisation présente l'avantage de répartir les vecteurs et les matrices des problèmes locaux sur les différentes mémoires. Mais le recouvrement des sous-domaines augmente la taille des données, et pour la méthode CMS le calcul de plusieurs modes sur chaque sous-domaine est très coûteux en temps de calcul et en mémoire. Par ailleurs, les méthodes CMS et FCMS réclament beaucoup de paramètres, ce qui les rend complexe à l'utilisation. En effet il est nécessaire de préciser la décomposition de domaine, le nombre de modes de flux et de courant sur chaque sous-domaine, les critères de convergence pour les résolutions locales et globale, et le tableau d'indices permettant de définir les fonctions lisses à ajouter pour la méthode FCMS. Un autre point faible de ces méthodes est qu'il semble difficile de prévoir l'erreur d'approximation en fonction des différents paramètres. En particulier la solution obtenue est très sensible à la décomposition de domaine choisie, et l'augmentation du nombre de modes ne suffit pas à compenser un mauvais choix. Cette sensibilité aux paramètres rend les méthodes CMS et FCMS peu robustes et difficilement utilisables dans un contexte industriel. Une contrainte supplémentaire est imposée par le solveur MINOS : les sous-domaines doivent être des rectangles. On a vu que ce type de sous-domaines est bien adapté pour des géométries cartésiennes telles que celle du REP, mais qu'il ne permet pas de traiter*

*correctement des géométries non cartésiennes comme celle du RJH. L'utilisation du solveur triangulaire en cours de développement ([Baudron et al., 2007]) autorisera des géométries beaucoup plus variées, et permettra sans doute d'améliorer la qualité d'approximation sur des cœurs tels que celui du RJH.*

*Au final, les méthodes CMS et FCMS présentent des perspectives intéressantes (présentées en fin de chapitre). Mais la facilité et l'efficacité de la parallélisation sont compensées par le coût élevé en temps de calcul et en mémoire. Leur sensibilité à plusieurs paramètres les rend difficile d'utilisation. Par ailleurs, l'intégration de la méthode CMS au projet APOLLO3/DESCARTES a nécessité une évolution du solveur MINOS pour permettre le calcul de plusieurs modes propres.*

*Pour ces raisons, nous avons souhaité développer une autre méthode de décomposition de domaine robuste, efficace en parallèle et qui limite au maximum les surcoûts en temps de calcul et en mémoire par rapport à un calcul direct MINOS. De plus on souhaite que l'intégration à APOLLO3/DESCARTES demande un minimum de modifications du programme existant. Nous avons choisi de développer un algorithme itératif, présenté au chapitre suivant.*

# Chapitre 7

# Description d'une méthode de décomposition de domaine itérative

---

*Notre objectif est de développer un algorithme itératif de décomposition de domaine inspiré de l'algorithme de Schwarz présenté à la section 3.2.2. Cette algorithme doit permettre de résoudre un problème à valeur propre afin de traiter le problème critique de la diffusion des neutrons sous forme mixte duale. Notre choix est d'utiliser des conditions de Robin aux interfaces des sous-domaines, avec une décomposition de domaine sans recouvrement de façon à limiter le nombre d'inconnues des problèmes locaux. A chaque itération, les valeurs du flux $\varphi$ et du courant $\overrightarrow{p}$ aux interfaces sont échangées afin d'y définir les conditions aux bords pour l'itération suivante.*

*La première partie du chapitre est consacrée à la description de l'algorithme, tandis que la deuxième partie présente des tests en 1D afin de valider la méthode et d'étudier son comportement sur des géométries hétérogènes.*

---

## 7.1 Adaptation de l'algorithme de Schwarz pour la résolution du problème de la diffusion mixte duale

Nous souhaitons utiliser l'algorithme de Schwarz pour la résolution du problème de la diffusion des neutrons. Le problème à source s'écrit :

$$\begin{array}{l} -\overrightarrow{\nabla}.(D\overrightarrow{\nabla}\varphi) + \sigma_a\varphi = S \quad \text{sur } R, \\ \varphi = 0 \quad \text{sur } \partial R. \end{array} \tag{7.1}$$

Supposons que le domaine $R$ soit décomposé en $K$ sous-domaines non recouvrants : $R = \bigcup_{k=1}^{K} R^k$, avec $\mathring{R}^k \cap \mathring{R}^l = \emptyset, 1 \leq k, l \leq K, k \neq l$. On note $\Gamma^{kl} = \partial R^k \cap \partial R^l$ les interfaces entre sous-domaines.

Le problème (7.1) est de la forme de (3.12). On peut donc envisager sa résolution par l'algorithme de Schwarz modifié (3.14) : la convergence avec les conditions du deuxième ordre (3.13) aux interfaces est assurée par le théorème 3.1. Mais nous souhaitons résoudre le problème de la diffusion sous sa forme mixte :

$$\left\{ \begin{array}{l} \overrightarrow{p} + D\overrightarrow{\nabla}\varphi = 0, \\ \overrightarrow{\nabla}.\overrightarrow{p} + \sigma_a\varphi = S \quad \text{sur } R, \\ \varphi = 0 \quad \text{sur } \partial R, \end{array} \right. \tag{7.2}$$

avec $D \geq C > 0$ et $\sigma_a \geq 0$. On remarque que la solution de ce problème nous fournit à la fois le flux $\varphi$ et le courant $\overrightarrow{p}$. Or, d'après la première équation de ce problème (loi de Fick (1.6)), le courant dérive du flux par la relation : $\overrightarrow{p} = -D\overrightarrow{\nabla}\varphi$. L'idée est donc d'utiliser des conditions de Robin aux interfaces :

$$-\overrightarrow{p_n^k}\cdot\overrightarrow{n^k} + \alpha^{kl}\varphi_n^k = \overrightarrow{p_{n-1}^l}\cdot\overrightarrow{n^l} + \alpha^{kl}\varphi_{n-1}^l \quad \text{sur } \Gamma^{kl}, \tag{7.3}$$

où $\alpha^{kl}$ est strictement positif, $n$ est l'indice d'itération et $\overrightarrow{n^k}$, $\overrightarrow{n^l}$ sont les normales sortantes sur $\Gamma^{kl}$ ($\overrightarrow{n^k} = -\overrightarrow{n^l}$). Ces conditions sont bien du type (3.13) avec $\beta^{kl} = 0$, car $\overrightarrow{p_n^k}\cdot\overrightarrow{n^k} = -D\dfrac{\partial\varphi_n^k}{\partial n^k}$. L'algorithme que nous proposons consiste donc à résoudre sur chaque sous-domaine $R^k$ et à chaque itération d'indice $n$ :

$$\left\{ \begin{array}{l} \overrightarrow{p_n^k} + D\overrightarrow{\nabla}\varphi_n^k = 0 \quad \text{sur } R^k, \\ \overrightarrow{\nabla}.\overrightarrow{p_n^k} + \sigma_a\varphi_n^k = S \quad \text{sur } R^k, \\ -\overrightarrow{p_n^k}\cdot\overrightarrow{n^k} + \alpha^{kl}\varphi_n^k = S^k_{\Gamma^{kl},n-1} \quad \text{sur } \Gamma^{kl}, \forall l \text{ tel que } \Gamma^{kl} \neq \varnothing, \\ \varphi_n^k = 0 \quad \text{sur } \partial R^k \cap \partial R, \end{array} \right. \tag{7.4}$$

avec :

$$S^k_{\Gamma^{kl},n-1} = \overrightarrow{p_{n-1}^l}\cdot\overrightarrow{n^l} + \alpha^{kl}\varphi_{n-1}^l \quad \text{sur } \Gamma^{kl}. \tag{7.5}$$

A chaque itération, un problème à source de diffusion est à résoudre sur chaque sous-domaine. Le but étant d'utiliser le solveur MINOS pour résoudre ces problèmes locaux, il est nécessaire de mettre ce problème sous forme variationnelle. La formulation variationnelle mixte duale est obtenue en multipliant les deux premières équations de (7.4) par des fonctions tests, en intégrant sur $R^k$ et en appliquant la formule de Green au deuxième terme du membre de gauche

de la première équation. Le problème obtenu est alors de trouver sur chaque sous-domaine $R^k$ : $\overrightarrow{p_n^k} \in H(div, R^k)$ et $\varphi_n^k \in L^2(R^k)$ solution de

$$\begin{cases} \displaystyle\int_{R^k} -\frac{1}{D}\overrightarrow{p_n^k}\cdot\overrightarrow{q} + \int_{R^k} \overrightarrow{\nabla}.\overrightarrow{q}\,\varphi_n^k - \sum_{l,\Gamma^{kl}\neq\emptyset} \frac{1}{\alpha^{kl}} \int_{\Gamma^{kl}} \left(\overrightarrow{p_n^k}\cdot\overrightarrow{n^k}\right)\left(\overrightarrow{q}\cdot\overrightarrow{n^k}\right) \\ = \displaystyle\sum_{l,\Gamma^{kl}\neq\emptyset} \frac{1}{\alpha^{kl}} \int_{\Gamma^{kl}} S^k_{\Gamma^{kl},n-1}\left(\overrightarrow{q}\cdot\overrightarrow{n^k}\right) \quad \forall \overrightarrow{q} \in H(div, R^k), \\ \displaystyle\int_{R^k} \overrightarrow{\nabla}.\overrightarrow{p_n^k}\psi + \int_{R^k} \sigma_a \varphi_n^k \psi = \int_{R^k} S\psi \quad \forall \psi \in L^2(R^k). \end{cases} \tag{7.6}$$

On remarque l'apparition de deux termes d'interface dans la première équation, qui correspondent aux conditions de Robin imposées sur $\Gamma^{kl}$. En particulier, le terme source fait intervenir $S^k_{\Gamma^{kl},n-1}$ défini par (7.5). Or, le calcul de $S^k_{\Gamma^{kl},n-1}$ nécessite de connaître les valeurs de $\overrightarrow{p^l_{n-1}}$ et de $\varphi^l_{n-1}$ sur $\Gamma^{kl}, \forall l$ tel que $\Gamma^{kl} \neq \varnothing$. La résolution de (7.6) nous fournissant un flux $\varphi^l_{n-1} \in L^2(R^l)$, $S^k_{\Gamma^{kl},n-1}$ n'est donc pas bien défini puisque $\varphi^l_{n-1}$ n'est pas défini sur $\Gamma^{kl}$. Pour contourner cette difficulté, nous introduisons une approximation numérique qui intervient dans la discrétisation des conditions aux interfaces.

### 7.1.1 Discrétisation des conditions aux interfaces avec l'élément de Raviart-Thomas

Dans l'optique de la résolution des problèmes locaux (7.6) par MINOS, le flux et le courant sont discrétisés à l'aide de l'élément de Raviart-Thomas introduit à la section 2.2.3. Ainsi le flux est polynomial par morceaux, avec des discontinuités aux interfaces entre mailles. Les degrés de liberté de flux correspondent aux valeurs du flux aux nœuds . Sans perte de généralité, nous nous plaçons dans le cas de l'élément $RT_0$, où le flux est constant sur chaque maille.

Au lieu de considérer les conditions aux interfaces "continues" (7.3), qui ne sont pas bien définies dans le cas de la formulation mixte duale puisque les flux sont dans $L^2$, on introduit des conditions approchées. Pour ce faire, on impose un recouvrement d'une maille entre les sous-domaines voisins. Ainsi, les nœuds de flux aux bords de deux sous-domaines voisins sont communs. L'idée est donc d'utiliser les valeurs du flux sur ces nœuds dans les conditions aux interfaces. Contrairement au flux, le courant est bien défini sur les bords. Comme on le voit sur la représentation 2.1a de l'élément $RT_0$, les nœuds de courant sont situés sur les bords de la maille. On utilise donc dans les conditions aux interfaces les valeurs du courant aux nœuds situés sur cette interface.

Soit $\Gamma_0^{kl}$ l'ensemble des nœuds de flux communs à $R^k$ et $R^l$, c'est à dire l'ensemble des nœuds de flux contenus dans le recouvrement d'une maille. On définit les conditions aux interfaces discrètes par :

$$-\hat{p}^k_{n|\Gamma^k} + \alpha^{kl}\hat{\varphi}^k_{n|\Gamma_0^{kl}} = \hat{p}^l_{n-1|\Gamma^k} + \alpha^{kl}\hat{\varphi}^l_{n-1|\Gamma_0^{kl}}. \tag{7.7}$$

Le "$\hat{}$" sur $p$ et $\varphi$ signifie que l'on considère les solutions discrètes. $\hat{p}^k_{n|\Gamma^k}$ et $\hat{p}^l_{n-1|\Gamma^k}$ correspondent aux valeurs des courants sur l'ensemble des nœuds de courant de $\Gamma^k = \partial R^k \cap \overline{R^l}$ : $\hat{p}^k_{n|\Gamma^k} = \overrightarrow{\hat{p}^k_n}\cdot\overrightarrow{n^k}$ sur $\Gamma^k$, et $\hat{p}^l_{n-1|\Gamma^k} = \overrightarrow{\hat{p}^l_{n-1}}\cdot\overrightarrow{n^l}$ sur $\Gamma^k$. $\hat{\varphi}^k_{n|\Gamma_0^{kl}}, \hat{\varphi}^l_{n-1|\Gamma_0^{kl}}$ correspondent aux valeurs des flux sur les nœuds de $\Gamma_0^{kl}$.

L'approximation vient donc du fait que les flux ne sont pas évalués sur l'interface. En effet, les nœuds de flux utilisés dans (7.7) sont internes à la maille, contrairement à ceux

de courant qui se situent sur l'interface. Les flux et les courants ne sont donc pas évalués aux mêmes nœuds . La figure 7.1 illustre l'utilisation des nœuds de l'élément $RT_0$ dans les conditions aux interfaces discrètes (7.7). Le domaine est décomposé en deux sous-domaines qui se recouvrent d'une maille. On distingue les nœuds sur $\Gamma^1, \Gamma^2$ et $\Gamma_0^{12}$. Plus le maillage est fin,

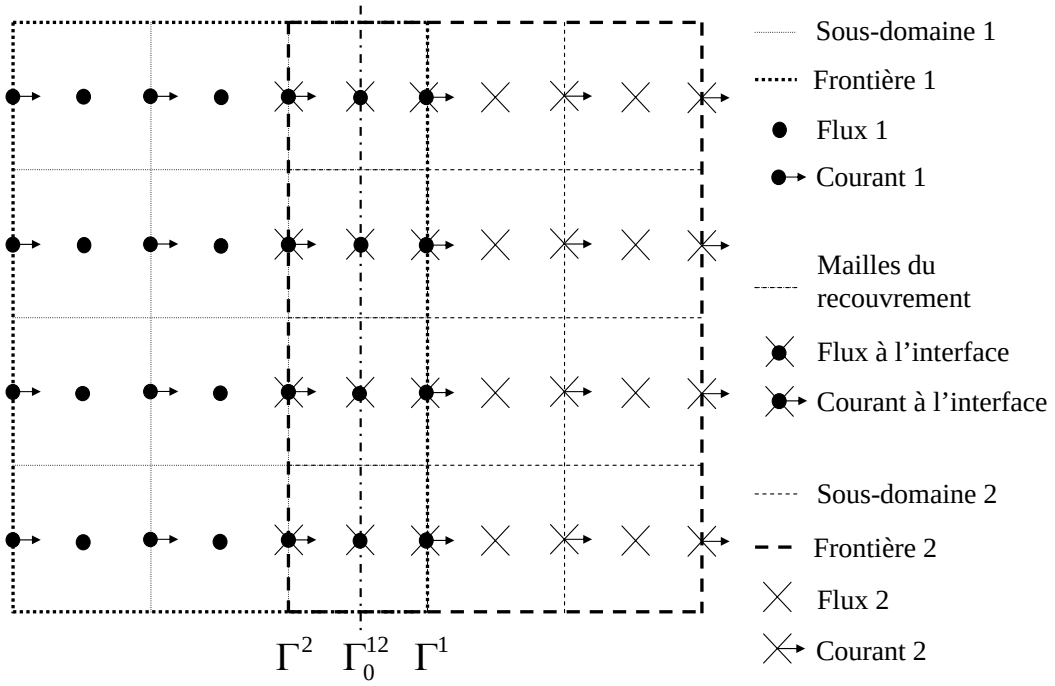



FIG. 7.1: Noeuds de l'élément $RT_0$ pour une décomposition en deux sous-domaines qui se recouvrent d'une maille.

plus le recouvrement est petit (sa taille étant d'une maille) et meilleure est l'approximation de la condition aux interfaces (7.3) par la condition (7.7). En effet plus la maille de recouvrement est petite et plus les nœuds de flux et de courant utilisés dans (7.7) sont proches.

Ces conditions aux interfaces discrètes interviennent dans l'algorithme (7.4) et dans sa formulation variationnelle (7.6) par le biais de la reformulation du terme source (7.5). Le terme source discrétisé est :

$$\hat{S}^k_{\Gamma^{kl},n-1} = \hat{p}^l_{n-1|\Gamma^k} + \alpha^{kl}\hat{\varphi}^l_{n-1|\Gamma_0^{kl}}. \tag{7.8}$$

### 7.1.2 Résolution du problème à valeur propre

Nous venons de décrire un algorithme qui permet de résoudre le problème à source de la diffusion sous forme mixte duale. Mais le problème qui nous intéresse est le problème critique (2.3), sans source externe. Or, nous avons vu à la section 1.2.2 que c'est un problème à valeur propre. On veut donc adapter l'algorithme (7.4) à la résolution de ce problème.

Nous avons décrit à la section 2.2.9 les méthodes numériques utilisées par le solveur MINOS pour résoudre le problème (2.3). En particulier, l'algorithme itératif des puissances permet de résoudre le problème à valeur propre. A chaque itération, cet algorithme résout un problème à source, puis met à jour la source et la valeur propre avant de passer à l'itération suivante.

L'idée est d'utiliser le même jeu d'itération pour l'algorithme de décomposition de domaine et pour l'algorithme des puissances. A chaque itération, un problème à source est résolu sur chaque sous-domaine $R^k$ en utilisant les conditions (7.7) aux interfaces, puis la source de fission $\hat{S}^k_{f,n-1}$ et la valeur propre $\hat{\lambda}_{n-1}$ sont mises à jour :

$$\begin{cases} \overrightarrow{\hat{p}^k_n} + D\overrightarrow{\nabla}\hat{\varphi}^k_n = 0 \quad \text{sur } R^k, \\ \overrightarrow{\nabla}.\overrightarrow{\hat{p}^k_n} + \sigma_a\hat{\varphi}^k_n = \dfrac{1}{\hat{\lambda}_{n-1}}\hat{S}^k_{f,n-1} \quad \text{sur } R^k, \\ -\hat{p}^k_{n|\Gamma^k} + \alpha^{kl}\hat{\varphi}^k_{n|\Gamma^{kl}_0} = \hat{S}^k_{\Gamma^{kl},n-1} \quad \text{sur } \Gamma^{kl}, \forall l \text{ tel que } \Gamma^{kl} \neq \varnothing, \\ \hat{\varphi}^k_n = 0 \quad \text{sur } \partial R^k \cap \partial R. \end{cases} \tag{7.9}$$

La formulation variationnelle de ce problème est :

$$\begin{cases} \displaystyle\int_{R^k} -\frac{1}{D}\overrightarrow{\hat{p}^k_n}\cdot\overrightarrow{\hat{q}} + \int_{R^k}\overrightarrow{\nabla}.\overrightarrow{\hat{q}}\,\hat{\varphi}^k_n - \sum_{l,\Gamma^{kl}\neq\emptyset}\frac{1}{\alpha^{kl}}\int_{\Gamma^{kl}}\left(\overrightarrow{\hat{p}^k_n}\cdot\overrightarrow{n^k}\right)\left(\overrightarrow{\hat{q}}\cdot\overrightarrow{n^k}\right) \\ = \displaystyle\sum_{l,\Gamma^{kl}\neq\emptyset}\frac{1}{\alpha^{kl}}\int_{\Gamma^{kl}}\hat{S}^{\Gamma^{kl}}_{n-1}\left(\overrightarrow{\hat{q}}\cdot\overrightarrow{n^k}\right), \\ \displaystyle\int_{R^k}\overrightarrow{\nabla}.\overrightarrow{\hat{p}^k_n}\hat{\psi} + \int_{R^k}\sigma_a\hat{\varphi}^k_n\hat{\psi} = \frac{1}{\hat{\lambda}_{n-1}}\int_{R^k}\hat{S}^k_{f,n-1}\hat{\psi}. \end{cases} \tag{7.10}$$

Tout l'enjeu est d'étudier la convergence de cette méthode de décomposition de domaine, appelée méthode IDD (*Iterative Domain Decomposition method*). En effet, une condition nécessaire pour qu'elle soit performante est que l'ajout de l'algorithme itératif de décomposition de domaine n'entraîne pas une forte augmentation du nombre d'itérations externes nécessaire pour la convergence.

## 7.2 Tests de la méthode IDD en 1D

Afin d'analyser la convergence de la méthode IDD pour la résolution du problème de la diffusion critique, nous commençons par une application numérique simple : géométrie 1D, un groupe d'énergie. L'efficacité de la méthode est évaluée par le nombre d'itérations nécessaires pour atteindre un critère d'arrêt donné sur les itérations externes. Il est également important de mesurer l'incidence de la valeur des coefficients aux interfaces $\alpha^{kl}$ sur la convergence de la méthode. Pour cela, la même valeur $\alpha$ est adoptée pour toutes les interfaces, et chaque test est réalisé avec deux valeurs de $\alpha$ différentes : $\alpha = 10^{-1}$ et $\alpha = 10^{-2}$. Par ailleurs, dans la méthode originale proposée par Pierre-Louis Lions pour un problème à source, on peut montrer que la convergence est d'autant plus lente que le nombre de sous-domaines est grand ([Nataf, 2001]). Afin de voir si ce phénomène se vérifie pour la méthode IDD, trois décompositions de domaine sont testées avec 3, 6 et 17 sous-domaines de tailles égales. La géométrie utilisée (Géométrie 1) est celle représentée à la figure 4.1, les sections efficaces étant décrites dans le tableau 4.1. Le tableau 7.1 donne pour les différentes configurations le nombre d'itérations nécessaire pour atteindre un critère d'arrêt de $10^{-6}$ sur la norme $L^2$ de la source pour les itérations externes.

La première constatation est que le nombre d'itérations augmente peu en utilisant la méthode IDD par rapport au calcul direct, quelque soit le nombre de sous-domaines. Deuxièmement, le nombre d'itérations augmente avec le nombre de sous-domaines, mais l'algorithme semble assez peu sensible à ce paramètre. Troisièmement, le choix $\alpha = 10^{-2}$ permet de diminuer significativement le nombre d'itérations par rapport à $\alpha = 10^{-1}$.

TAB. 7.1: Nombre d'itérations nécessaire pour atteindre un critère d'arrêt de $10^{-6}$. $k_{eff} = 1.008238$ (Géométrie 1).

| | Calcul direct | 3 sous-domaines | 6 sous-domaines | 17 sous-domaines |
|---|---|---|---|---|
| $\alpha = 10^{-2}$ | 143 | 206 | 214 | 222 |
| $\alpha = 10^{-1}$ | 143 | 219 | 245 | 313 |

Afin d'évaluer l'influence des sections efficaces, nous présentons des tests avec les sections d'absorption et de fission multipliées par 5 par rapport aux tests précédents (Géométrie 2). Le flux varie alors plus fortement, comme on peut le voir sur la figure 7.2. Les nombres d'itérations pour les différentes configurations sont présentés dans le tableau 7.2.

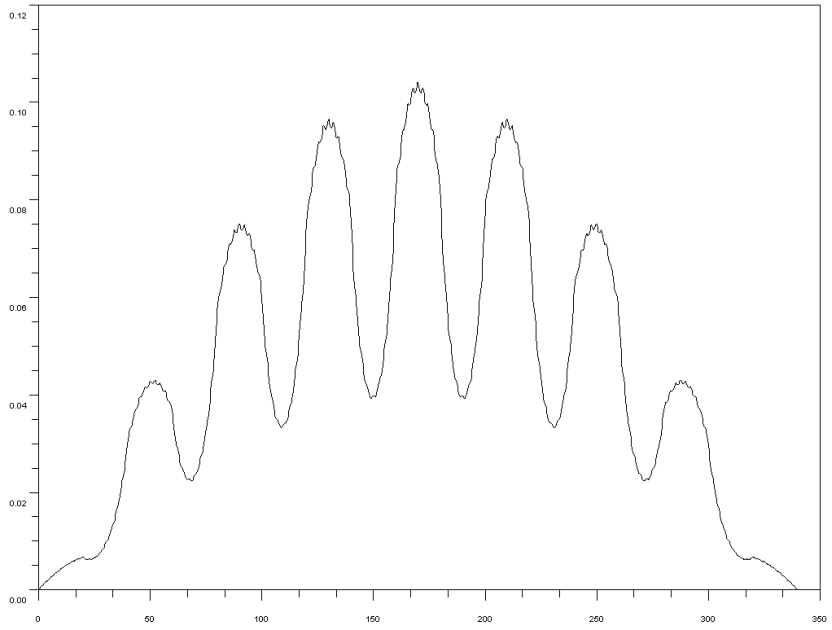



FIG. 7.2: Représentation du flux sur la géométrie 2. $k_{eff} = 1,0825794$ (Géométrie 2).

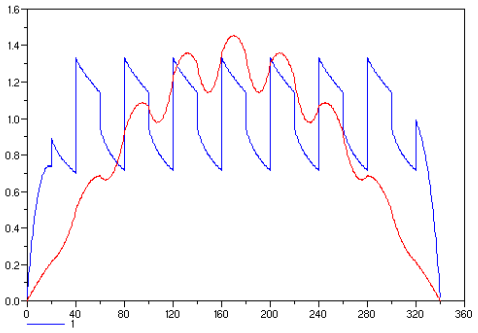
TAB. 7.2: Nombre d'itérations nécessaire pour atteindre un critère d'arrêt de $10^{-6}$.

| | Calcul direct | 3 sous-domaines | 6 sous-domaines | 17 sous-domaines |
|---|---|---|---|---|
| $\alpha = 10^{-2}$ | 812 | 789 | 710 | 600 |
| $\alpha = 10^{-1}$ | 812 | 894 | 955 | 1001 |

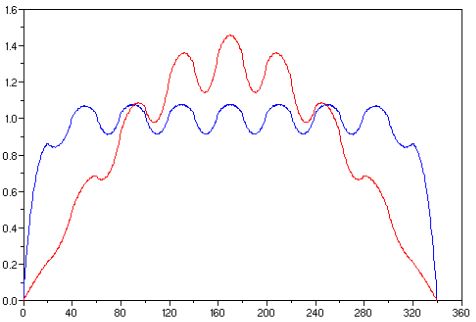
Le nombre d'itérations nécessaire pour le calcul direct passe de 143 à 812. Pour la méthode IDD dans le cas $\alpha = 10^{-1}$, ce nombre augmente un peu, et d'autant plus que le nombre de sous-domaines est grand. Par contre, la situation s'inverse avec $\alpha = 10^{-2}$ : le nombre d'itérations est moins élevé avec la méthode IDD qu'avec la résolution directe, et diminue quand le nombre de sous-domaines augmente. La décomposition avec 17 sous-domaines, qui

correspond à un découpage selon les assemblages, est celle qui demande le moins d'itérations avec $\alpha = 10^{-2}$. Or le flux varie fortement aux interfaces entre assemblages. Il semble donc que la décomposition de domaine, via les conditions aux interfaces, facilite la convergence de la méthode des puissances.

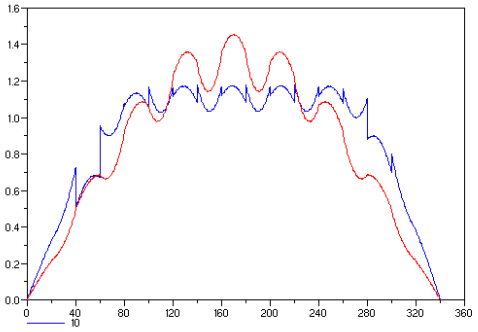
Afin de comparer la convergence de l'algorithme direct et celle de la méthode IDD, les flux sont représentés sur la figure 7.3 pour les deux méthodes aux itérations 10, 20 et 40. La courbe inchangée représente le flux convergé. On remarque que l'algorithme direct décrit l'allure locale du flux très rapidement. Par contre les variations globales demandent plus d'itérations pour être correctement approchées. Comme nous l'avons vu, la méthode IDD mélange la convergence de la décomposition de domaine et celle du problème à valeur propre. Les grandes discontinuités du flux à la première itération sont dues au mauvais raccord des solutions locales. Mais aux itérations 10 et 20, on constate que les sauts du flux et de sa dérivée s'estompent, jusqu'à disparaître à l'itération 40. Dans le même temps, la convergence du problème à valeur propre a bien lieu, et il semble qu'à l'itération 40 la méthode IDD soit plus proche de la solution convergée que le calcul direct.



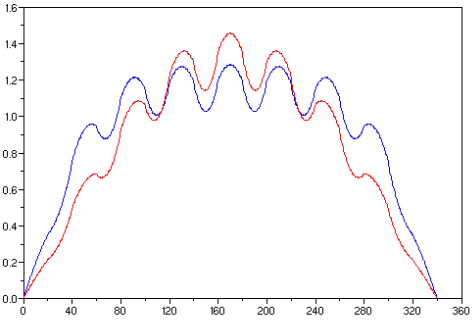

(a) 1 itération



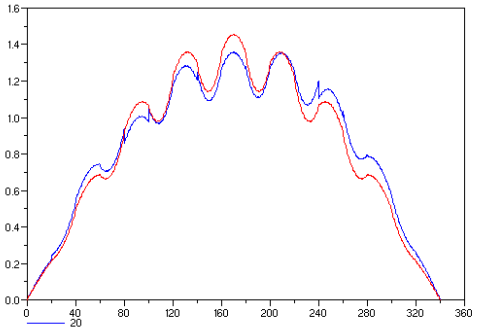
(b) 1 itération



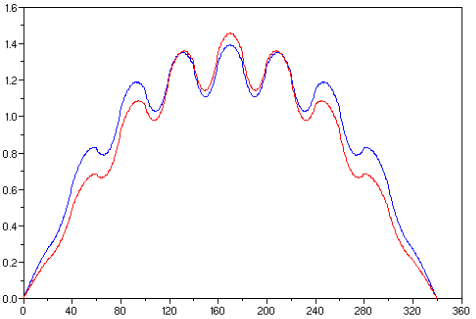

(c) 10 itérations



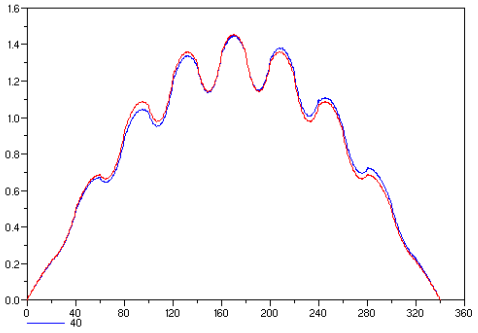
(d) 10 itérations



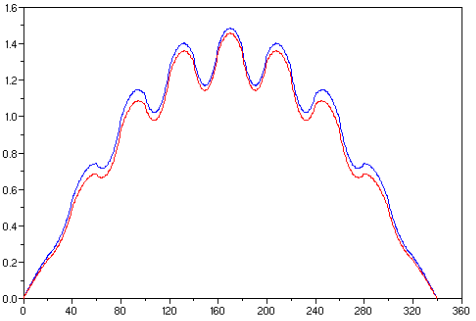

(e) 20 itérations



(f) 20 itérations




(g) 40 itérations



(h) 40 itérations

FIG. 7.3: Représentation des flux au cours des itérations. A droite la convergence du calcul direct, à gauche la convergence de la méthode IDD avec 17 sous-domaines.

---

*Nous avons développé une méthode itérative de décomposition de domaine (IDD) qui permet de résoudre le problème à valeur propre de la diffusion mixte duale. L'algorithme proposé dérive de l'algorithme de Schwarz avec des conditions de Robin aux interfaces. L'originalité de la méthode est d'utiliser un seul jeu d'itérations pour faire converger simultanément la décomposition de domaine et la résolution du problème à valeur propre par la méthode des puissances. Une approximation doit être faite au niveau des conditions aux interfaces, car le flux est dans $L^2$ et n'est donc pas défini sur les bords des sous-domaines.*

*Les tests 1D montrent que le nombre d'itérations n'augmente pas beaucoup lorsque l'on passe d'un calcul direct à un calcul avec la méthode IDD, et peut même diminuer lorsque la géométrie est fortement hétérogène. La convergence de la méthode ne semble pas trop sensible au nombre de sous-domaines et aux coefficients présents dans les conditions de Robin.*

*Aux vues de ces premiers résultats, la méthode IDD semble donc performante. Dans le chapitre suivant, nous allons voir si les tests sur des géométries complexes en 2D et 3D confirment cette impression, et si la méthode est efficace en parallèle.*

---

# Chapitre 8

# Applications numériques et parallélisation de la méthode IDD

---

*Notre but est d'appliquer la méthode IDD à des géométries complexes et réalistes : le REP 900 MWe et le RJH, en 2D et 3D. La convergence de l'algorithme est étudiée, notamment pour vérifier si les résultats encourageants obtenus en 1D sont valables en 2D et 3D sur des domaines fortement hétérogènes. Pour cela, nous avons implémenté l'algorithme dans le projet APOLLO3/DESCARTES, avec la bibliothèque MPI pour les communications entre processus. Les résolutions des problèmes locaux sont assurées par le solveur MINOS. Des tests de performance et d'efficacité sur plusieurs processeurs sont présentés, ainsi que les écarts avec un calcul direct MINOS sur le $k_{eff}$ et la puissance afin d'évaluer la qualité de l'approximation.*

---

## 8.1 Programmation de la méthode IDD pour le solveur MINOS

La méthode IDD est programmée de manière à être la plus transparente possible vis-à-vis de l'utilisateur. La classe *Subdomain* permet de réaliser l'ensemble des opérations sur un sous-domaine. En effet le constructeur de la classe permet de :

- créer deux vecteurs par interface, afin de stocker les valeurs des sources aux interfaces à chaque itération. Le premier vecteur contient les valeurs à transmettre aux sous-domaines voisins, tandis que le deuxième reçoit celles à récupérer ;
- lancer le calcul local avec le solveur MINOS ;
- réaliser les échanges de message avec les sous-domaines voisins.

La classe *Subdomain* fait appel a d'autres classes :

- la classe *Subdomain_ Solver*, qui construit tous les objets nécessaires au solveur MINOS afin de réaliser les calculs locaux. Ces objets sont décrits dans la section 2.3.1 ;
- la classe Subdomain_Util. Entre autres elle construit les maillages des sous-domaines.

La méthode telle que nous l'avons programmée ne demande que peu de paramètres à l'utilisateur :

- les valeurs des coefficients $\alpha^{kl}$. Ces coefficients sont définis pour chaque interface et pour chaque groupe d'énergie ;
- la décomposition du domaine dans chaque direction. Le nombre de sous-domaines est égal au produit du nombre d'intervalles choisi pour chaque axe. Les sous-domaines sont alors des rectangles de dimension égale, de façon à répartir équitablement la charge de calcul.

Les autres paramètres sont liés au solveur MINOS lui-même : nombre d'itérations internes, nombre d'itérations externes maximum, critères d'arrêt sur la source et le $k_{eff}$...

Il suffit donc pour l'utilisateur d'appeler le constructeur de la classe *Subdomain* avec les bons paramètres. Il obtient en sortie un objet *Mi_ Flux* contenant le flux, le courant et le $k_{eff}$ après convergence de la méthode.

### 8.1.1 La parallélisation de la méthode IDD

La parallélisation est prise en compte de manière native dans la programmation de la méthode IDD. En effet chaque objet *Subdomain* est attribué à un processus distinct. Ainsi, quelque soit le nombre de processeurs dont on dispose, autant de processus que de sous-domaines sont créés.

La parallélisation impose une synchronisation des itérations externes de tous les solveurs locaux. Seulement deux phases de communication sont nécessaires à chaque itération. La première échange les sources aux interfaces entre processus gérant des sous-domaines voisins. La deuxième rassemble les produits scalaires sur les sources de fission locales nécessaires à l'algorithme des puissances. Ces produits scalaires permettent ensuite de calculer un $k_{eff}$ global commun à tous les sous-domaines. Cette communication s'effectue de tous les processus vers tous les processus à l'aide d'un *MPI_ Allreduce*. De cette manière les produits scalaires globaux sont calculés pendant la communication $MPI$ à partir de tous les produits scalaires locaux.

Le schéma 8.1 présente les communications entre deux processeurs au travers des différentes étapes réalisées par le solveur MINOS à chaque itération. Les communications des sources aux interfaces se font de manière non bloquante : elles débutent à la fin des résolutions des systèmes linéaires réalisées sur chaque sous-domaine et à chaque itération, et doivent se terminer avant les résolutions des systèmes linéaires de l'itération suivante. L'opération de réduction sur

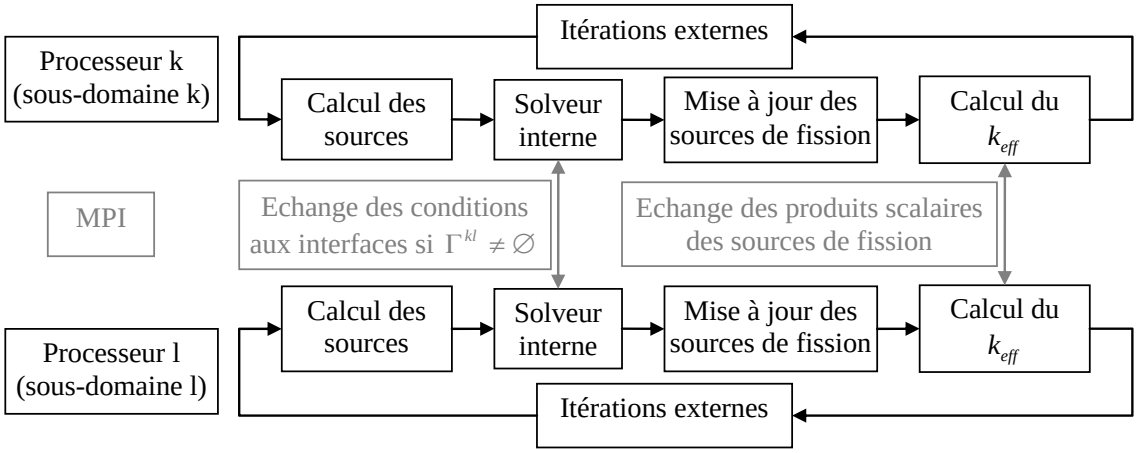



FIG. 8.1: Schéma de la parallélisation de la méthode IDD.

les produits scalaires est par contre bloquante, car la fonction *MPI_Allreduce* demande une synchronisation de tous les processus.

Un des avantages de la méthode IDD est qu'elle n'entraîne que peu de modifications du solveur MINOS. En effet seul l'appel aux deux fonctions de la classe *Subdomain* qui assurent les communications est à ajouter à la classe *EigenValueSolver*. Une modification est également à apporter à la classe *MatWspn* de façon à prendre en compte dans les matrices $W_d$ du système (2.42) les conditions de Robin aux interfaces. Cette modification vient du terme sur l'interface $\Gamma^{kl}$ présent dans le membre de gauche de la première équation du problème (7.10). Elle est simplifiée par la numérotation des inconnues de courant, expliquée à la section 2.2.7. En effet dans chaque direction $d$, les interfaces ne correspondent qu'aux premiers et derniers nœuds de courant. De plus l'intégrale sur $\Gamma^{kl}$ n'affecte que les termes diagonaux de la matrice $W_d$ correspondant aux nœuds de $\Gamma^{kl}$. La contribution des conditions d'interface ne modifie donc que les premiers et derniers termes de la diagonale de $W_d$.

### 8.1.2 Changement de l'estimateur de convergence de MINOS

Dans la section 2.2.9, nous avons décrit la méthode des puissances et les critères utilisés par MINOS pour sortir des itérations externes. Mais le critère sur la source de fission (2.46) pose problème pour deux raisons. La première est que si $S^n \equiv 0$, l'estimateur de convergence est infini. Or ce cas peut arriver avec la méthode IDD, sur un sous-domaine où la section de fission est nulle partout (par exemple si le sous-domaine ne contient que du réflecteur). La deuxième raison est que si l'on prend par exemple le vecteur $\bar{S}_i^{n+1} \equiv 0$, et $\bar{S}_i^n$ avec un seul élément non nul égal à 1, l'estimateur est égal à la taille $N$ des vecteurs, et explose si $N$ est grand.

On choisit donc de changer d'estimateur afin de répondre à ces problèmes, et on adopte le critère d'arrêt suivant :

$$\frac{\max\limits_{1\leq i\leq N} |\bar{S}_i^{n+1} - \bar{S}_i^n|}{\max\left(\max\limits_{1\leq i\leq N} |\bar{S}_i^n|, \max\limits_{1\leq i\leq N} |\bar{S}_i^{n+1}|\right)} \leq \epsilon_S. \tag{8.1}$$

Afin d'éviter une division par 0, l'estimateur renvoie 0 si le dénominateur est nul.

Pour la méthode IDD, des maximums locaux sont calculés sur chaque sous-domaine. Au moment de l'échange des produits scalaires sur les sources, une opération de réduction globale sur ces maximums (*MPI_ Allreduce*) permet de calculer les maximums globaux qui interviennent dans (8.1). De cette manière, tous les processus utilisent le même estimateur global, et sortent donc de la boucle sur les itérations externes au même nombre d'itérations. Dans tous les tests qui suivent, nous avons opté pour une borne supérieure $\epsilon_S = 10^{-5}$.

## 8.2 Convergence et performance de la méthode IDD sur des géométries complexes 2D et 3D

### 8.2.1 Le REP 3D

Le domaine, la géométrie et le maillage du REP 3D que nous utilisons sont décrits à la section 5.2.2. Le nombre de plans est ici porté à 40, avec une taille de maille égale à une cellule sur chaque plan. Le nombre de mailles est donc égal à 3 340 840. Le calcul de diffusion utilise deux groupes d'énergie et l'élément $RT_0$. Les coefficients des conditions aux interfaces (les $\alpha^{kl}$ dans (7.7)) sont choisis de manière à assurer la convergence : ils sont égaux quelque soit l'interface, mais ils varient en fonction du groupe d'énergie. On choisit $\alpha^{kl} = 1$ pour le groupe rapide, et $\alpha^{kl} = 5 \times 10^{-2}$ pour le groupe thermique, $\forall 1 \leq k, l \leq K, k \neq l$.

Le tableau 8.1 présente les résultats de la méthode IDD : nombre d'itérations, comparaison avec un calcul direct MINOS convergé, temps de calcul et efficacité en parallèle. La première

TAB. 8.1: Résultats de la méthode IDD pour un calcul REP 3D. $k_{eff} = 1,05208$.

| | Nombre d'itérations | $\Delta k_{eff}$ (pcm) | $\|\Delta P\|_2$ ($\times 10^{-4}$) | $\|\Delta P\|_\infty$ ($\times 10^{-3}$) | Temps réel (s.) | Efficacité par itération (%) |
|---|---|---|---|---|---|---|
| MINOS | 249 | 11 | 7,7 | 2,2 | 331 | 100 |
| 2 (2, 1, 1) | 247 | 10 | 7,7 | 2,2 | 188 | 87 |
| 4 (2, 2, 1) | 252 | 10 | 7,6 | 2,2 | 102 | 82 |
| 6 (3, 2, 1) | 250 | 29 | 21 | 3,3 | 80 | 69 |
| 8 (2, 2, 2) | 253 | 10 | 11 | 2,8 | 65 | 65 |
| 9 (3, 3, 1) | 251 | 47 | 25 | 4,4 | 61 | 61 |
| 12 (4, 3, 1) | 250 | 60 | 26 | 4,2 | 40 | 69 |
| 16 (4, 4, 1) | 251 | 73 | 28 | 3,6 | 32 | 65 |
| 18 (3, 3, 2) | 251 | 47 | 26 | 5,3 | 34 | 55 |

ligne du tableau fait référence au calcul direct MINOS ; toutes les autres lignes correspondent à la méthode IDD avec différents nombres de sous-domaines. Dans la première colonne figurent entre parenthèses les nombres par lesquelles on décompose respectivement les axes $x, y, z$. Le nombre de sous-domaines est donné par le produit de ces trois nombres. La deuxième colonne donne le nombre d'itérations nécessaire pour atteindre le critère de convergence (8.1) avec $\epsilon_S = 10^{-5}$. Les colonnes 3 à 5 présentent les écarts relatifs avec un calcul MINOS complètement convergé (10000 itérations externes). La colonne 3 donne l'écart sur le $k_{eff}$, la colonne 4 l'écart de puissance en norme 2 et la colonne 5 en norme infinie. Les colonnes 6 et 7 permettent d'évaluer les performances en parallèle. Le calcul MINOS est effectué en séquentiel, tandis que le nombre de processeurs utilisés par la méthode IDD est égal au nombre de sous-

domaines. La colonne 6 présente le temps réel écoulé entre le début et la fin de l'exécution du code, et la colonne 7 donne l'efficacité par itération de la méthode IDD, une efficacité de 100% étant attribuée au MINOS séquentiel.

La première remarque est que les résultats 1D en ce qui concerne le nombre d'itérations sont confirmés. En effet ce nombre n'augmente pas lorsque le nombre de sous-domaines croît. Cela se traduit naturellement au niveau des temps de calcul, qui diminuent fortement avec l'augmentation du nombre de sous-domaines. On passe de 331 secondes pour MINOS à 32 secondes avec la méthode IDD et 16 sous-domaines. L'efficacité est comprise entre 50% et 90%. Cela est dû au fait que la création du domaine et de la géométrie globale, ainsi que le calcul des sections efficaces macroscopiques sont effectués par tous les processeurs. La parallélisation de ces étapes permettraient d'avoir des efficacité meilleures, sans doute proches de 100%.

La deuxième remarque est que la qualité d'approximation, évaluée par les écarts avec le calcul MINOS convergé à 10000 itérations, est particulièrement bonne lorsque la décomposition de domaine respecte la symétrie du cœur (3 plans de symétrie qui partagent chaque axe en deux). En effet, les écarts avec 2, 4, 8 sous-domaines sont du même ordre que ceux du calcul MINOS utilisant le même critère d'arrêt. Cela s'explique par le fait que l'approximation au niveau des conditions aux interfaces est meilleure quand les interfaces correspondent aux axes de symétrie. En effet, sur ces axes le courant est faible, et le flux est plat. Le fait d'évaluer le courant et le flux sur des nœuds proches mais différents constitue donc une bonne approximation d'un nœud commun sur l'interface. Cette approximation se dégrade si le courant est fort et si le flux varie rapidement au niveau des interfaces, ce qui est le cas pour les décompositions en 6, 9, 12, 16, 18 sous-domaines. Cependant l'écart relatif de puissance en norme infinie est dans tous les cas plus petit que 0,6%, et l'écart sur le $k_{eff}$ infèrieur à 80pcm. La figure 8.2 représente la distribution de l'écart de puissance sur le vingtième plan entre la solution de référence et la solution fournie par la méthode IDD avec 6 sous-domaines. Les deux sauts visibles sur cette figure correspondent aux deux interfaces verticales, confirmant l'erreur d'approximation faite sur les conditions à ces interfaces. L'interface horizontale respecte la symétrie du domaine, et ne fait pas apparaître de discontinuité dans l'écart de puissance.

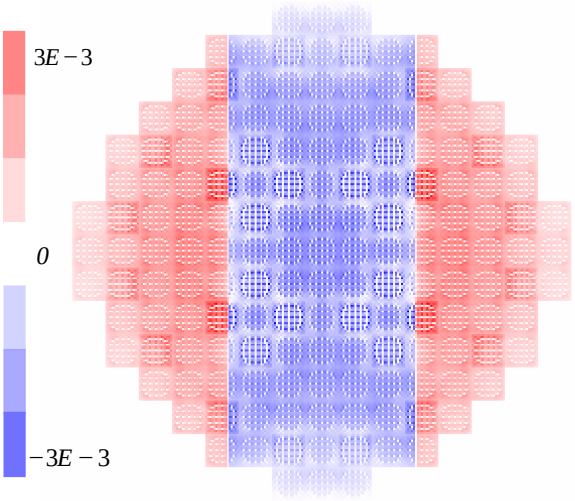



FIG. 8.2: Représentation graphique de la distribution de l'écart de puissance sur le vingtième plan entre la solution de référence et la solution fournie par la méthode IDD avec 6 sous-domaines.

**Analyse de la qualité d'approximation sur le REP 2D**

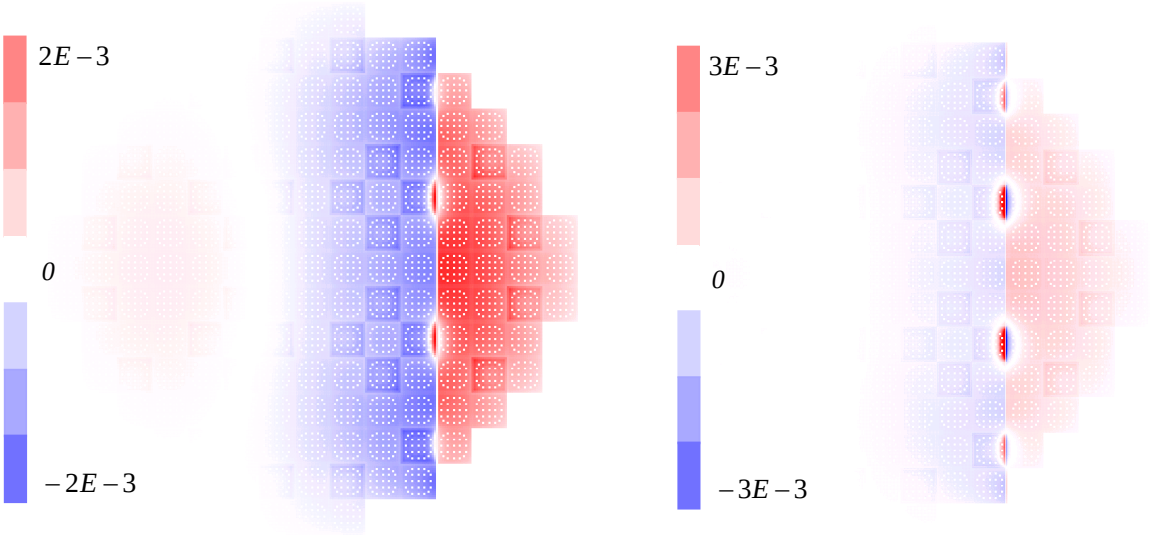
Nous souhaitons vérifier numériquement sur le REP 2D que l'approximation par la méthode IDD s'améliore lorsque la taille des mailles aux interfaces diminue (voir section 7.1.1). On choisit une décomposition de domaine qui ne correspond pas aux axes de symétrie du cœur : deux sous-domaines sont utilisés, partagés par un axe vertical correspondant à douze assemblages de largeur d'un côté et cinq de l'autre. Quatre interfaces UOX/MOX sont présentes sur cette axe. Deux maillages sont utilisés : le premier a 289 mailles dans chaque direction, le deuxième en a 1156 (quatre fois plus).

Le tableau 8.2 donne pour les deux maillages les écarts relatifs sur le $k_{eff}$ et sur la puissance entre un calcul direct et un calcul IDD, tous les deux convergés à 10000 itérations externes. Il apparaît que l'écart sur le $k_{eff}$ et la norme 2 de l'écart de puissance sont nettement plus faibles avec le maillage plus fin. Par contre, la norme infinie de l'écart de puissance est de façon étonnante un peu plus élevée. Les cartes des écarts de puissance représentées sur la figure 8.3 donnent une explication à ce résultat. On voit en effet que l'écart est partout plus faible dans le cas du maillage fin, sauf aux interfaces entre assemblages UOX et MOX situés sur l'interface entre les deux sous-domaines. Sur ces interfaces, l'approximation ne s'améliore pas en diminuant le pas du maillage, car les variations du flux restent très fortes entre deux mailles. Les histogrammes 8.4a et 8.4b confirment ces répartitions des écarts de puissance : environ 79% des mailles ont un écart plus petit que 0,1% pour le maillage grossier, alors que ce pourcentage passe à 99% pour le maillage fin.

Le choix d'un axe où les variations du flux sont moins fortes aurait probablement permis d'obtenir une amélioration plus nette de l'approximation, surtout en norme infinie. On voit donc que les écarts mesurés sur le REP 3D avec des décompositions de domaine qui ne correspondent pas aux axes de symétrie du cœur , auraient été plus faibles avec un maillage plus fin.

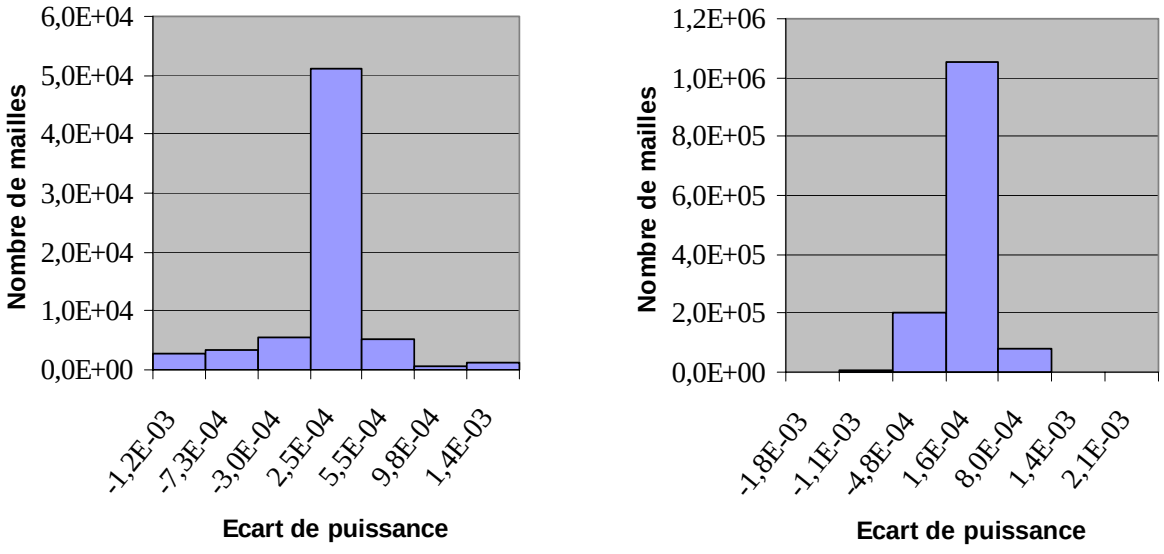
TAB. 8.2: Ecarts relatifs sur la puissance et sur le $k_{eff}$ entre la méthode IDD avec deux sous-domaines et MINOS pour le REP 2D. $k_{eff} = 1,097$.

| | 289 mailles | 1156 mailles |
|---|---|---|
| $\Delta k_{eff}$ (pcm) | 14,0 | 3,5 |
| $\|\Delta P\|_2$ | $1,4 \times 10^{-3}$ | $3,8 \times 10^{-4}$ |
| $\|\Delta P\|_\infty$ | $2,0 \times 10^{-3}$ | $3,0 \times 10^{-3}$ |




(a) 289 mailles par direction (b) 1156 mailles par direction

FIG. 8.3: Représentations graphiques des distributions de l'écart de puissance entre la méthode IDD avec deux sous-domaines et MINOS pour le REP 2D.




(a) 289 mailles par direction (b) 1156 mailles par direction

FIG. 8.4: Histogrammes représentant le nombre de mailles en fonction de l'écart de puissance entre la méthode IDD avec deux sous-domaines et MINOS pour le REP 2D.

### 8.2.2 Le RJH 2D

Nous utilisons le même domaine, la même géométrie et le même maillage que ceux décrits dans la section 6.2.2. Le calcul utilise le modèle de la diffusion à six groupes d'énergie et l'élément $RT_0$. Les coefficients des conditions aux interfaces sont choisis pour assurer la convergence de la méthode : $\alpha^{kl} = 10^{-2}$, quelques soient l'interface et le groupe d'énergie.

Le tableau 8.3 est similaire au tableau 8.1, mais concerne maintenant le calcul du RJH 2D. Le calcul MINOS complètement convergé utilise 50000 itérations externes.

TAB. 8.3: Résultats de la méthode IDD pour un calcul RJH 2D. $k_{eff} = 1,30857$.

| | Nombre d'itérations | $\Delta k_{eff}$ (pcm) | $\|\Delta P\|_2$ ($\times 10^{-3}$) | $\|\Delta P\|_\infty$ ($\times 10^{-2}$) | Temps réel (s.) | Efficacité par itération (%) |
|---|---|---|---|---|---|---|
| MINOS | 941 | 60 | 4,2 | 1,9 | 1288 | 100 |
| 2 $(2,1)$ | 914 | 60 | 4,3 | 2,0 | 671 | 93 |
| 4 $(2,2)$ | 925 | 60 | 4,3 | 1,9 | 303 | 104 |
| 6 $(3,2)$ | 932 | 56 | 4,3 | 2,0 | 254 | 84 |
| 8 $(4,2)$ | 903 | 50 | 4,3 | 1,9 | 140 | 110 |
| 9 $(3,3)$ | 941 | 50 | 4,4 | 2,0 | 139 | 103 |
| 12 $(4,3)$ | 887 | 45 | 4,4 | 2,0 | 85 | 119 |
| 16 $(4,4)$ | 869 | 45 | 4,3 | 2,0 | 62 | 120 |
| 20 $(5,4)$ | 931 | 45 | 4,3 | 1,9 | 55 | 116 |
| 25 $(5,5)$ | 926 | 44 | 4,6 | 2,0 | 49 | 103 |
| 30 $(6,5)$ | 912 | 48 | 4,6 | 2,0 | 36 | 116 |
| 36 $(6,6)$ | 916 | 49 | 4,5 | 2,0 | 29 | 120 |

Ces tests sur le RJH 2D illustrent de la même manière que ceux sur le REP 3D le fait que la décomposition de domaine n'augmente pas le nombre d'itérations. La méthode IDD permet même dans tous les cas d'atteindre le critère de convergence avec moins d'itérations. D'autre part, on constate que la méthode IDD apporte une approximation équivalente à MINOS à même nombre d'itérations. Ce n'était pas le cas pour les tests sur le REP 3D, sauf avec une décomposition de domaine respectant les axes de symétrie. Pourtant le RJH ne présente pas d'axe de symétrie, mais le maillage du RJH est beaucoup plus fin que celui du REP, ce qui améliore l'approximation des conditions aux interfaces. Au niveau des performances, les temps de calcul diminuent de manière très importante lorsque l'on augmente le nombre de sous-domaines. De 1288 secondes pour un calcul direct par MINOS, on passe à 29 secondes avec 36 sous-domaines. De même que pour le REP, les constructions du domaine, de la géométrie et des sections ne sont pas parallélisées. Mais elles sont beaucoup plus rapides car elles se font de manière plus simple et plus directe que pour le REP 3D. Les efficacités sont en conséquence excellentes, la plupart du temps plus grandes que 100%. Lorsque l'efficacité est supérieure à 100%, cela veut dire que pour $N$ processeurs utilisés, à même nombre d'itérations, l'algorithme IDD est plus de $N$ fois plus rapide qu'un calcul direct MINOS. Ce résultat paraît étonnant et mérite d'être analysé.

### Temps d'un calcul MINOS par rapport au nombre d'inconnues

La complexité de l'ensemble des opérations réalisées par le solveur MINOS est linéaire par rapport au nombre de mailles, notamment parce que le profil des matrices ne dépend pas de ce nombre. Il est alors surprenant d'obtenir une efficacité rapportée au nombre d'itérations plus grande que 100% en faisant de la décomposition de domaine.

Afin de vérifier la complexité linéaire, nous présentons sur la figure 8.5 les résultats de tests où on évalue le temps de calcul en fonction du nombre de mailles. La géométrie du RJH est projetée sur des maillages de taille variable, et un calcul de diffusion à six groupes est réalisé, en fixant le nombre d'itérations externes à 100. Le temps de calcul mesuré correspond au temps de construction et de résolution du système linéaire. Il ne prend pas en compte les constructions du domaine, de la géométrie et du maillage.

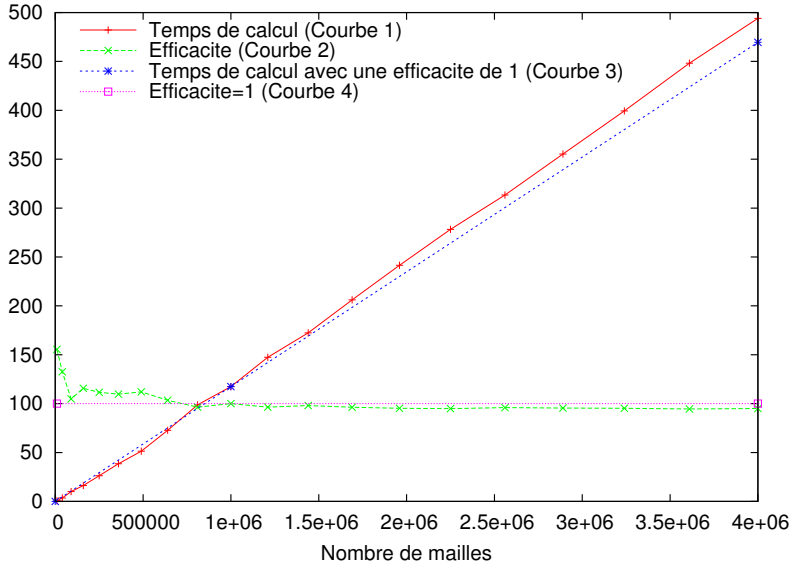



FIG. 8.5: Temps d'un calcul MINOS en fonction du nombre de mailles.

La courbe 1 représente les temps de calcul en fonction du nombre de mailles utilisé. La courbe 3 correspond à des temps de calcul proportionnels au nombre de mailles, en prenant comme référence un million de mailles. On remarque que la courbe 1 n'est pas linéaire : avec moins de un million de mailles elle est en-dessous de la courbe 3, avec plus de un million de mailles elle est au-dessus. Les temps de calcul augmentent donc un peu plus que proportionnellement au nombre de mailles.

La courbe 2 représente l'"efficacité" associée au nombre de mailles, c'est à dire le temps de calcul rapporté au nombre de mailles, par rapport au calcul de référence à un million de mailles. La comparaison avec la courbe 4 (droite représentant une efficacité de 100%) montre bien que cette efficacité est supérieure à 100% pour moins de un million de mailles, infèrieure à 100% pour plus de un million de mailles.

La partie du graphique qui correspond aux nombres de mailles des calculs sur les sous-domaines de la méthode IDD se situe entre 2500 et 500000 mailles pour des décompositions de domaine allant de 2 à 36 sous-domaines. Dans cet intervalle, la courbe 2 donne une efficacité variant entre 105% et 140%, ce qui explique les efficacités plus grandes que 100% dans le tableau 8.3.

### Convergence de MINOS sur le RJH

On remarque dans le tableau 8.3 que le nombre d'itérations demandé par MINOS pour atteindre le critère de convergence (8.1) avec $\epsilon_S = 10^{-5}$ est élevé. Or on constate que même après 941 itérations, les écarts relatifs avec un calcul MINOS convergé (50000 itérations) sont assez importants : 60pcm pour le $k_{eff}$, et 2% pour l'écart maximum de puissance. La figure 8.6 représente la distribution de l'écart de puissance. Pour le REP 3D, la convergence est beaucoup plus rapide, puisque après 249 itérations les écarts avec un calcul convergé avec 10000 itérations sont de 11pcm pour le $k_{eff}$, et de 0.2% pour l'écart maximum de puissance (voir le tableau 8.1).

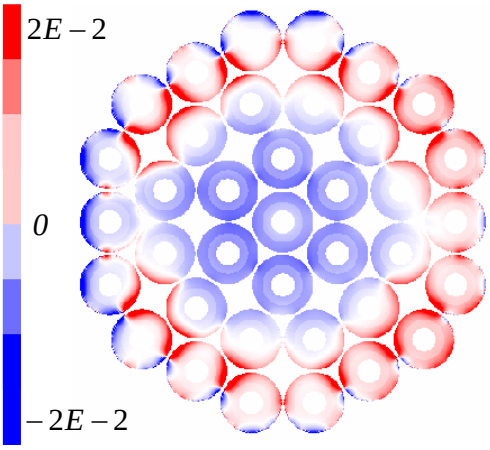



FIG. 8.6: Représentation graphique de la distribution de l'écart de puissance sur le RJH 2D entre un calcul MINOS à 941 itérations, et un calcul MINOS à 50000 itérations.

Cette lenteur de convergence ne dépend pas du nombre de mailles utilisé, comme l'attestent des tests réalisés avec des maillages différents. De même l'utilisation de modèles $SP_3$ ou $SP_5$ à la place de la diffusion n'améliore pas la convergence. Plusieurs raisons peuvent expliquer cette difficulté de l'algorithme à converger.

D'une part, 6 groupes d'énergie sont utilisés alors que le calcul du REP n'a que deux groupes. Comme la résolution sur les groupes d'énergie est assurée par une méthode de Gauss-Seidel à une itération, il est normal que l'augmentation du nombre de groupes se répercute sur le nombre d'itérations externes.

D'autre part, la géométrie du RJH est très hétérogène, non cartésienne et sans axe de symétrie. Il semble que dans ce cas, la résolution interne sur les directions assurée par une méthode de Gauss-Seidel ne soit pas très efficace. L'utilisation de plus d'une itération interne ne permet d'ailleurs pas d'améliorer la convergence globale. Le remplacement de cette méthode par une méthode de Krylov telle que le gradient conjugué permettrait sans doute d'accélérer la convergence.

Enfin, il semble que l'accélération de Tchebychev de la méthode des puissances ne soit pas très performante dans le cas du RJH. Cette accélération est réalisée sur les sources de fission. On pourrait l'appliquer aux vecteurs de flux pour tous les groupes, ce qui serait probablement plus efficace. L'inconvénient est que l'accélération de Tchebychev nécessite le stockage des vecteurs des deux itérations précédentes. Il faudrait donc stocker les vecteurs de flux pour tous les groupes, alors que la source de fission est unique, ce qui implique un surcoût en mémoire allouée.

### 8.2.3 Premier calcul d'un RJH en 3D

Nous souhaitons maintenant tester notre méthode IDD sur une géométrie 3D du RJH. Un calcul avec un maillage fin en 3D du cœur complet du RJH n'a jamais été réalisé auparavant, il est donc nécessaire dans un premier temps de construire le domaine de calcul. Le cœur mesure 120cm de haut. Il est décomposé axialement en 3 zones : la zone centrale est la partie combustible, celles du haut et du bas sont équivalentes, constituées d'eau et de béryllium (réflecteur). La zone combustible mesure 60cm de hauteur, divisée en 14 mailles (dix de 5cm et quatre de 2,5cm). La géométrie de chacun des plans est la même que la géométrie 2D présentée à la section 2.3.2. Les zones haute et basse se décomposent en deux parties. La première partie contient uniquement de l'eau, mesure 20cm et est divisée en quatre mailles de 5cm. La deuxième partie, composée d'eau et de béryllium, mesure 10cm décomposé en une maille de 5cm et deux mailles de 2,5cm. Le cœur est donc décomposé en 28 mailles axiales. La figure 8.7 représente cette décomposition axiale du RJH 3D.

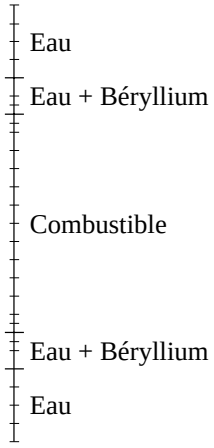



FIG. 8.7: Maillage axial du RJH 3D.

Dans les axes $x$ et $y$, 1000 mailles sont utilisées par axe, ce qui porte le nombre total de mailles à 28 millions. Le calcul utilise le modèle de la diffusion avec 6 groupes d'énergie et l'élément $RT_0$. Les coefficients aux interfaces sont les mêmes qu'en 2D. Le nombre d'itérations externes est fixé à 1000.

Ce calcul est très coûteux, à la fois en temps CPU et en mémoire. Son exécution demandant plus de 32 Go de mémoire vive, nous avons choisis la machine Lucretia décrite à la section 3.1.2 pour réaliser nos tests. Le tableau 8.4 donne les temps de calcul et les efficacités obtenus avec la méthode IDD avec 4 et 6 sous-domaines, avec comme référence le calcul direct séquentiel MINOS. Le nombre de processeurs est égal au nombre de sous-domaines, et la décomposition de domaine est réalisée en découpant les axes $x, y, z$ par les nombres entre parenthèses dans la première colonne. La valeur du $k_{eff}$ est donnée dans la deuxième colonne.

L'efficacité des calculs en parallèle est bonne, de l'ordre de 70%. Ce résultat est satisfaisant, d'autant plus que le calculateur n'était pas dédié à ces calculs. Les communications n'ont donc pas trop d'impact sur l'efficacité, alors que les échanges des conditions aux interfaces représentent un grand nombre d'inconnues. Ainsi, l'utilisation de 6 processeurs permet de réduire le temps de calcul par plus de quatre. Quant aux valeurs du $k_{eff}$, leurs écarts ne dépassent pas 10pcm, ce qui laisse penser que la convergence est très bonne, bien qu'une solution de référence ne soit pas disponible pour le prouver.

TAB. 8.4: Résultats de la méthode IDD pour un calcul RJH 3D. $k_{eff} = 1,13938$.

| | $k_{eff}$ | Temps réel (s.) | Efficacité par itération |
|---|---|---|---|
| MINOS | 1,13938 | 63613 | 100 |
| 4 processeurs (2,2,1) | 1,13938 | 20832 | 76 |
| 6 processeurs (3,2,1) | 1,13947 | 15513 | 68 |

### 8.2.4 Perspectives pour la méthode IDD

Un point important est que les coefficients $\alpha^{kl}$ aux interfaces sont pour le moment choisis pour assurer une bonne convergence. Mais cette façon de procéder n'est pas satisfaisante pour deux raisons. La première est que plusieurs calculs doivent être exécutés par l'utilisateur afin de trouver des valeurs satisfaisantes des $\alpha^{kl}$. La deuxième raison est que ces coefficients ne sont pas optimisés. En effet il est possible de les faire varier sur chaque interface et pour chaque groupe, ce que nous n'avons pas fait. Nous envisageons deux possibilités pour calculer de manière automatique et optimisée ces coefficients. La première possibilité est de s'inspirer des travaux exposés dans [Nataf, 2001] pour déterminer des conditions aux interfaces optimisées. Mais la détermination du coefficient optimisé utilise des transformations de Fourier des équations, ce qui n'est pas possible dans notre cas. Cependant, la démarche permettrait de se faire une idée de coefficients optimisés sur des cas simplifiés, et d'étudier notamment la dépendance de ces coefficients aux pas du maillage et aux sections efficaces. La deuxième possibilité serait de calculer les $\alpha^{kl}$ comme un rapport d'un courant sur un flux déterminé à partir d'un calcul grossier sur tout le cœur .

Si la détermination automatique et optimisée des coefficients aux interfaces s'avère efficace, une perspective de la méthode IDD serait de fournir une méthode générale pour la parallélisation des solveurs déterministes du projet APOLLO3/DESCARTES. En effet, la généralisation de la méthode aux équations $SP_N$ ne devrait pas poser de problèmes, et les boucles sur les harmoniques sont déjà présentes dans le code que nous avons développé. Par contre l'application au modèle du transport n'est pas évidente à priori.

La méthode IDD permet d'envisager des couplages de modèles. Par exemple des sous-domaines où le flux varie beaucoup, ou qui présentent un intérêt particulier (pic de puissance, réflecteur...), peuvent utiliser le modèle $SP_3$, tandis que les autres sous-domaines font intervenir le modèle de la diffusion. Les couplages peuvent se faire par des matrices d'interpolation, ou par la méthode des joints.

L'utilisation de maillages différents sur les sous-domaines est également une possibilité offerte par la méthode IDD, alors que le solveur MINOS ne le permet pas. Ainsi, on peut utiliser un maillage plus fin sur les sous-domaines où l'on souhaite avoir une meilleure approximation. Les conditions aux interfaces doivent alors utiliser des matrices d'interpolation afin de prendre en compte les différentes tailles de maille.

Mais l'utilisation de modèles ou de maillages différents pose le problème de la répartition des charges de calcul en parallèle. En effet, l'usage d'un maillage et d'un modèle unique garantit une bonne répartition des calculs sur des sous-domaines de même taille. A l'inverse, un modèle plus complexe ou un maillage plus fin doivent être compensés par un sous-domaine plus petit. L'équilibrage des charges semble assez complexe à assurer. Or un déséquilibre est synonyme d'une perte d'efficacité, car les processus doivent se synchroniser à la fin de chaque itération pour réaliser les communications globales.

---

*Les exemples numériques sur des géométries complexes en 2D et 3D que nous venons de présenter montrent que la méthode IDD permet d'obtenir la solution d'un problème critique de diffusion multigroupe avec une très bonne approximation, en conservant un nombre d'itérations semblable à celui d'un calcul direct MINOS. Cela se traduit par une très bonne efficacité de la méthode en parallèle, proche ou supérieure à 100%. L'adoption de sous-domaines de même taille permet d'avoir une bonne répartition des tâches, et d'éviter l'attente de certains processeurs à chaque itération au moment des communications bloquantes. Les plus gros échanges de données concernent les communications des conditions aux interfaces. Mais ces échanges ne sont pas bloquants, et peuvent recouvrir jusqu'à une itération de calcul. La méthode IDD permet également de répartir les données sur les différentes mémoires disponibles. En effet les matrices et les vecteurs du système linéaire sur un sous-domaine sont stockés dans la mémoire du processeur chargé de la résolution du problème local sur ce sous-domaine.*

*Le développement d'une méthode permettant de déterminer automatiquement des coefficients de Robin optimisés aux interfaces reste à faire. A cette condition, et après un protocole de validation, la méthode IDD semble être suffisamment robuste, efficace et facile d'utilisation pour un usage industriel.*

---

# Conclusion et perspectives

Notre travail de thèse a consisté à développer deux méthodes de décomposition de domaine pour la résolution du problème à valeur propre de la diffusion sous forme mixte duale. Ces méthodes ont toutes deux été programmées en C++ dans le cadre du projet APOLLO3/DESCARTES, et ont été conçues pour utiliser le solveur de cœur MINOS.

La première méthode (CMS) est une méthode de synthèse modale. Nous avons fait le choix d'une décomposition de domaine avec des sous-domaines recouvrants : outre la convergence d'ordre infini, le recouvrement permet d'éviter la mise en œuvre complexe de modes d'interface. La méthode a été adaptée pour résoudre des problèmes mixtes. Ainsi deux espaces d'approximation sont engendrés par des modes propres calculés par MINOS sur les sous-domaines, et prolongés par zéro sur le cœur . L'application au problème de la diffusion mixte duale a montré la nécessité d'employer des conditions de courant nul sur les bords internes des sous-domaines. Par ailleurs nous avons mis en évidence le fait que le nombre de modes de courant doit être plus élevé que le nombre de modes de flux. Des tests sur des géométries hétérogènes ont permis de voir que le choix de la décomposition de domaine a une grande influence sur la qualité d'approximation de la méthode, et doit prendre en compte la géométrie du domaine. Ainsi, une bonne approximation sur la puissance et le $k_{eff}$ est obtenue lorsque les bords des sous-domaines correspondent aux axes de symétrie des assemblages. Par ailleurs un recouvrement important des sous-domaines permet de limiter le nombre de modes à calculer.

Basée sur un principe de factorisation du flux, la méthode FCMS modifie les fonctions de base de la méthode CMS. Elle ne demande que le calcul du premier mode propre sur chaque sous-domaine. Les résultats numériques montrent qu'à même qualité d'approximation, la méthode FCMS permet de réduire les coûts en temps de calcul et en mémoire.

Les méthodes CMS et FCMS présentent deux atouts principaux. Premièrement, les résolutions des problèmes locaux sont totalement indépendantes. La parallélisation est donc simple, et les tests de la méthode FCMS en parallèle ont permis d'obtenir une bonne efficacité lorsque le nombre de processeurs n'est pas trop grand. De plus, cette indépendance des calculs locaux permet d'envisager l'usage de maillages et de modèles différents (diffusion, $SP_N$, transport) sur les sous-domaines, ce qui n'est pas possible avec le solveur MINOS. Une technique de type "base réduite" permettrait de limiter les résolutions de problèmes locaux à quelques sous-domaines. Le deuxième atout de ces méthodes est que les résolutions locales constituent une étape indépendante de la construction et de la résolution du système linéaire. Il est donc envisageable de constituer une "bibliothèque" de solutions locales utilisable pour plusieurs problèmes. En cinétique, les calculs locaux pourraient servir pour plusieurs pas de temps.

Mais la mise en œuvre de ces méthodes n'est pas évidente. Trop de paramètres sont à déterminer par l'utilisateur : nombre de modes de flux et de courant, critères de convergence, choix des caractéristiques des sous-domaines telles que taille, recouvrement, configuration.

De plus la qualité de l'approximation est très sensible à ces paramètres, et est donc difficile à prévoir.

La deuxième méthode (IDD) que nous avons développée est un algorithme itératif de type Schwarz avec des conditions de Robin aux interfaces. Bien que l'algorithme original soit conçu pour résoudre un problème à source, nous l'avons adapté à un problème mixte à valeur propre. Une approximation est faite au niveau des conditions aux interfaces pour faire intervenir à la fois le flux et le courant. Les itérations externes assurent la convergence simultanée de la décomposition de domaine et du problème à valeur propre. Les applications numériques montrent que le nombre d'itérations est proche de celui d'un calcul direct par MINOS, avec une sensibilité assez faible au nombre de sous-domaines et aux valeurs des coefficients de Robin aux interfaces. Il en résulte une excellente efficacité en parallèle, dans certains cas supérieure à 100%, et qui ne décroît pas lorsque le nombre de processeurs augmente. De plus l'algorithme semble robuste : il converge rapidement sur tous les tests que nous avons réalisés.

La méthode IDD est simple d'utilisation car elle ne demande que trois paramètres : le nombre de sous-domaines, le critère de convergence et les coefficients de Robin aux interfaces. La mise au point d'une technique pour déterminer des coefficients de Robin optimisés permettrait d'améliorer encore la convergence de l'algorithme, sa robustesse et sa facilité d'utilisation. Comme pour les méthodes CMS et FCMS, des modèles et des maillages différents peuvent être utilisés sur les sous-domaines, bien que la répartition de charge en parallèle semble difficile à gérer dans ce cas.

La méthode IDD est donc une méthode de décomposition de domaine performante pour la résolution du problème à valeur propre de la diffusion. Son extension à la résolution des équations $SP_N$ est simple à réaliser car les boucles sur les harmoniques sont déjà programmées dans le code. Robuste, performante et simple pour l'utilisateur, elle nous semble appropriée pour un futur développement industriel.

# Table des figures

# Liste des tableaux

# Bibliographie


[Allaire et Bal, 1999] G. Allaire et G. Bal. Homogenization of the critically spectral equation in neutron transport. *RAIRO modélisation mathématique et analyse numérique*, 33 :721–746, 1999.

[Allaire et Capdeboscq, 2000] G. Allaire et Y. Capdeboscq. Homogenization of a spectral problem in neutronic multigroup diffusion. *Computer Methods in Applied Mechanics and Engineering*, 187 :91–117, 2000.

[Babuska et Osborn, 1991] I. Babuska et J. Osborn. Eigenvalue problems. Dans P.G. Ciarlet et J. L. Lions, éditeurs, *Handbook of Numerical Analysis*, volume 2. Elsevier Science, 1991. Finite Element Methods (Part 1).

[Baudron *et al.*, 2007] A. M. Baudron, C. Döderlein, P. Guerin, J. J. Lautard, et F. Moreau. Unstructured 3D core calculations with the descartes system. application to the jhr research reactor. Dans *Proceedings of the Joint International Topical Meeting on Mathematics and Computation and Supercomputing in Nuclear Applications*, Monterey, Etats-Unis, avril 2007.

[Baudron et Lautard, 2007] A. M. Baudron et J.J. Lautard. MINOS : a simplified $P_N$ solver for core calculation. *Nuclear Science and Engineering*, 155 :250–263, 2007.

[Baudron *et al.*, 1999] A. M. Baudron, J. J. Lautard, et D. Schneider. Mixed dual methods for neutronic reactor core calculations in the CRONOS system. Dans *Proceedings of the American Nuclear Society Topical Meeting on Mathematical Methods and Supercomputing in Nuclear Applications*, Madrid, Espagne, avril 1999.

[Both *et al.*, 2003] J. P. Both, A. Mazzolo, Y. Peneliau, O. Petit, B. Roesslinger, Y. K. Lee, et M. Soldevila. Tripoli4, a monte-carlo particles transport code. main features and large scale application in reactor physics. Dans *Proceedings of the International Conference on Supercomputing in Nuclear Application*, Paris, France, Septembre 2003.

[Bourquin, 1989] F. Bourquin. Synthèse modale d'opérateurs elliptiques du second ordre. *Compte rendu de l'Académie des Sciences*, 309 :919–922, 1989. Série 1, N ° 16.

[Bourquin, 1992] F. Bourquin. Component mode synthesis and eigenvalues of second order operators : Discretization and algorithm. $M^2AN$, 26 :385–423, 1992.

[Brezzi *et al.*, 1987] F. Brezzi, J. Douglas Jr, M. Fortin, et L. D. Marini. Efficient rectangular mixed finite element in two and three space variables. $M^2AN$, 21(4) :581–604, 1987.

[Brezzi et Fortin, 1991] F. Brezzi et M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer Series in Computational Mathematics. Springer-Verlag, 1991.

[Brézis, 1983] H. Brézis. *Analyse fonctionnelle.* Collection Mathématiques appliquées pour la maîtrise. Masson, 1983.

[Bussac et Reuss, 1985] J. Bussac et P. Reuss. *Traité de neutronique.* Collection enseignement des sciences. Hermann, 1985.

[Calvin, 2005] C. Calvin. Descartes : A new generation system for neutronic calculations. Dans *Proceedings of the Joint International Topical Meeting on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, Avignon, France, septembre 2005.

[Charpentier *et al.*, 1995] I. Charpentier, F. De Vuyst, et Y. Maday. A component mode synthesis method of infinite order of accuracy using subdomain overlapping for polygonal domains. Dans *Proceedings of the ENUMATH*, Paris, 1995.

[Charpentier *et al.*, 1996] I. Charpentier, F. De Vuyst, et Y. Maday. Méthode de synthèse modale avec une décomposition de domaine par recouvrement. *Compte rendu de l'Académie des Sciences*, 322 :881–888, 1996. Série 1.

[Charpentier *et al.*, 1998] I. Charpentier, F. De Vuyst, et Y. Maday. The overlapping component mode synthesis method : the shifted eigenmodes strategy and the case of selfadjoint operators with discontinuous coefficients. Dans P. E. Bjorstad, M. S. Espedal, et D. E. Keyes, éditeurs, *Proceedings of the Ninth International Conference on Domain Decomposition Methods*, 1998.

[Charpentier *et al.*, 1999] I. Charpentier, Y. Maday, et A. T. Patera. Bounds evaluation for outputs of eigenvalue problems approximated by the overlapping modal synthesis method. *Compte rendu de l'Académie des Sciences*, 329 :909–914, 1999. Série 1.

[Ciarlet, 1982] P. G. Ciarlet. *Introduction à l'analyse numérique matricielle et à l'optimisation.* Collection Mathématiques appliquées pour la maîtrise. Masson, 1982.

[Collino *et al.*, 2000] F. Collino, S. Ghanemi, et P. Joly. Domain decomposition methods for harmonic wave propagation : a general propagation. *Computer Methods in Applied Mechanics and Engineering*, 184(2-4) :171–211, 2000.

[Coulomb, 1988] F. Coulomb. Domain decomposition and mixed finite elements for the neutron diffusion equation. Dans *Proceedings of the Second International Symposium on Domain Decomposition Methods*, Los Angeles, Etats-Unis, Janvier 1988. SIAM.

[Coulomb, 1989] F. Coulomb. *Algorithmes parallèles pour la résolution de l'équation de diffusion par des méthodes elements finis et par les méthodes nodales.* PhD thesis, Université Paris 6, 1989.

[Coulomb, 1990] F. Coulomb. Domain decomposition and associate block-Jacobi method for the diffusion equation. Dans *Proceedings of the Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphie, Etats-Unis, 1990. SIAM.

[Coulomb et Fedon-Magnaud, 1988] F. Coulomb et C. Fedon-Magnaud. Mixed and mixed-hybrid elements for the diffusion equation. *Nuclear Science and Engineering*, 100 :218–225, 1988.

[Coulomb et Lautard, ] F. Coulomb et J. J. Lautard. Comparaison numérique des éléments finis mixtes et des éléments finis de lagrange. Publication du Commissariat à l'Energie Atomique DEMT 87/111 SERMA/LENR/87-885 "T", CEA.

[Craig et Bampton, 1968] R. Craig et M. C. C. Bampton. Coupling of substructures for dynamic analysis. *American Institute of Aeronautics and Astronautics*, 6 :1313–1321, 1968.

[Dautray et Lions, 1988] R. Dautray et J. L. Lions. *Analyse mathématique et calcul numérique pour les sciences et les techniques.* INSTN Collection Enseignement. Masson, 1988. 9 volumes.

[Després, 1993] B. Després. Domain decomposition method and the helmholtz problem ii. Dans *Second International Conference on Mathematical and Numerical Aspects of Wave Propagation*, pages 197–206, Philadelphie, 1993. SIAM.

[Després *et al.*, 1992] B. Després, P. Joly, et J. E. Roberts. A domain decomposition method for the harmonic maxwell equations. Dans *Iterative Methods in Linear Algebra*, pages 475–484, Pays-Bas, Amsterdam, 1992.

[Guerin *et al.*, 2007a] P. Guerin, A. M. Baudron, J.J. Lautard, et S. Van Criekingen. Component mode synthesis methods for 3-D heterogeneous core calculations applied to the mixed-dual finite element solver MINOS. *Nuclear Science and Engineering*, 155 :264–275, 2007.

[Guerin *et al.*, 2005] P. Guerin, A. M. Baudron, et J. J. Lautard. A component mode synthesis method for 3d cell-by-cell $SP_N$ calculation unsing the mixed dual finite element solver MINOS. Dans *Proceedings of the Joint International Topical Meeting on Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, Avignon, France, Septembre 2005.

[Guerin *et al.*, 2006] P. Guerin, A. M. Baudron, et J. J. Lautard. Component mode synthesis methods applied to 3D heterogeneous core calculations, using the mixed dual finite element solver MINOS. Dans *Proceedings of the American Nuclear Society's Topical Meeting on Reactor Physics*, Vancouver, Canada, Septembre 2006.

[Guerin *et al.*, 2007b] P. Guerin, A. M. Baudron, et J. J. Lautard. Domain decomposition methods for core calculations using the MINOS solver. Dans *Proceedings of the Joint International Topical Meeting on Mathematics and Computation and Supercomputing in Nuclear Applications*, Monterey, Etats-Unis, avril 2007.

[Hurty, 1965] W. C. Hurty. Dynamic analysis of structural systems using component modes. *American Institute of Aeronautics and Astronautics*, 4 :678–685, 1965.

[Lathuiliere, ] Bruno Lathuiliere. *Décomposition de domaine, couplage multi-modèles et raffinement de maillage pour la simulation neutronique haute performance.* PhD thesis, Université Bordeaux 1. A paraître.

[Lautard, 1981] J. J. Lautard. New finite element representation for 3D reactor calculations. Dans *Proceedings of the International Topical Meeting on Advances in Mathematical Methods for the Solution of Nuclear Engineering Problems*, München, Allemagne, Avril 1981.

[Lautard, 1994] J. J. Lautard. La méthode nodale de CRONOS : MINOS. approximation par des éléments mixtes duaux. note interne CEA-N-2763, CEA, août 1994.

[Lautard et Flumiani, 2003] J. J. Lautard et T. Flumiani. Extension of the mixed dual finite element method to the solution of the $SP_N$ transport equation in 2D unstructured geometries composed by arbitrary quadrilaterals. Dans *Proceedings of the 5th International Conference on Supercomputing in Nuclear Applications*, Paris, France, septembre 2003. Organization for economic cooperation and development/Nuclear energy agency.

[Lautard et Moreau, 1993] J. J. Lautard et F. Moreau. A fast 3D parallel solver based on the mixed dual finite element approximation. Dans *Proceedings of the American Nuclear Society Topical Meeting on Mathematical Methods and Supercomputing in Nuclear Applications*, Karlsruhe, Allemagne, avril 1993.

[Lions, 1990] P. L. Lions. On the Schwarz alternating method III : A variant for nonoverlapping subdomains. Dans *Proceedings of the Third International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphie, Etats-Unis, 1990. SIAM.

[Loubière *et al.*, 1999] S. Loubière, R. Sanchez, et I. Zmijarevic. Apollo2 : twelve years later. Dans *Proceedings of the International Conference on Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications*, Madrid, Espagne, Septembre 1999.

[Nataf, 2001] F. Nataf. Interface connections in domain decomposition methods. Dans *NATO Advanced Study Institute*, volume 75 de *Modern Methods in Scientific Computing and Applications*. Université de Montréal, 2001. NATO Science Ser.II.

[Nataf et Rogier, 1995] F. Nataf et F. Rogier. Factorization of the convection-diffusion operator and the schwarz algorithm. *Mathematical Models and Methods in Applied Sciences*, 5(1) :67–93, 1995.

[Nataf *et al.*, 1994] F. Nataf, F. Rogier, et E. De Sturler. Optimal interface conditions for domain decomposition methods. note interne, CMAP (Ecole Polytechnique), 1994.

[Nedelec, 1986a] J. C. Nedelec. Mixed finite elements in $\mathbb{R}^3$. *Numerische Mathematik*, 50 :57–81, 1986.

[Nedelec, 1986b] J. C. Nedelec. A new family of mixed finite element in $\mathbb{R}^3$. *Numerische Mathematik*, 50 :57–81, 1986.

[Pinchedez, 1999] K. Pinchedez. *Calcul parallèle pour les équations de diffusion et de transport homogènes en neutronique*. PhD thesis, Université de Paris 6, 1999.

[Planchard, 1995] J. Planchard. *Méthodes mathématiques en neutronique*. Collection de la Direction des Etudes et Recherches d'Electricité de France. Eyrolles, 1995.

[Pomraning, 1993] G. J. Pomraning. Asymptotic and variational derivations of the simplified $P_N$ equations. *Annals Nuclear Energy*, 20(9) :623–637, 1993.

[Raviart et Thomas, 1988] P.A. Raviart et J.M. Thomas. *Introduction à l'analyse numérique des équations aux dérivées partielles*. Collection Mathématiques appliquées pour la maîtrise. Masson, 1988.

[Raviart et Thomas, 1977] P. A. Raviart et J. M. Thomas. A mixed finite element method for second order elliptic problems. *Lectures Notes in Mathematics*, 606 :275–291, 1977.

[Reuss, 1998] P. Reuss. *La Neutronique*. Collection Que sais-je ? Presses Universitaires de France, 1998.

[Reuss, 2003] P. Reuss. *Précis de neutronique*. Collection Génie atomique. EDP Sciences, 2003.

[Roberts et Thomas, 1991] J. E. Roberts et J. M. Thomas. Mixed and hybrid methods. Dans P. G. Ciarlet et J. L. Lions, éditeurs, *Handbook of Numerical Analysis*, volume 2. Elsevier Science, 1991. Finite Element Methods (Part 1).

[Saad, 2003] Y. Saad. *Iterative Methods for Sparse Linear System*. SIAM, 2003.

[Schneider, 2000] D. Schneider. *Eléments finis mixtes duaux pour la résolution numérique de l'équation de la diffusion neutronique en géométrie hexagonale*. PhD thesis, Université de Paris 6, decembre 2000.

[Schneider et Lautard, 2001] D. Schneider et J. J. Lautard. Mixed dual finite element methods for the treatment of hexagonal geometries in diffusion equation. Dans *Proceedings of the American Nuclear Society Topical Meeting on Mathematical Methods and Supercomputing in Nuclear Applications*, Salt Lake City, Utah, Etats-Unis, septembre 2001.

[Siess, 2004] V. Siess. *Homogénéisation des équations de criticité en transport neutronique*. PhD thesis, Université de Paris 6, 2004.

[Varga, 1962] R. S. Varga. *Matrix Iterative Analysis*. Prentice-Hall series in automatic computation. Prentice-Hall, 1962.

[Verdu *et al.*, 2005] G. Verdu, D. Ginestar, R. Miro, et V. Vidal. Using the jacobi-davidson method to obtain the dominant lambda modes of a nuclear power reactor. *Annals of Nuclear Energy*, 32 :1274–1296, 2005.

[Wachspress, 1966] E. L. Wachspress. *Iterative Solution of Elliptic System*. Prentice-Hall international series in applied mathematics. Prentice-Hall, 1966.