



HAL
open science

Développement et comparaison de bases d'éléments finis pour la résolution de l'équation de transport sur des maillages hexagonaux

David Labeurthre

► **To cite this version:**

David Labeurthre. Développement et comparaison de bases d'éléments finis pour la résolution de l'équation de transport sur des maillages hexagonaux. Physique mathématique [math-ph]. Université Grenoble Alpes [2020-..], 2022. Français. NNT : 2022GRALI090 . tel-04048632

HAL Id: tel-04048632

<https://theses.hal.science/tel-04048632v1>

Submitted on 28 Mar 2023

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

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

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ GRENOBLE ALPES

École doctorale : I-MEP2 - Ingénierie - Matériaux, Mécanique, Environnement, Energétique, Procédés, Production

Spécialité : MEP : Mécanique des fluides Energétique, Procédés

Unité de recherche : CEA SACLAY, Laboratoire de Logiciels pour la Physique des Réacteurs

Développement et comparaison de bases d'éléments finis d'ordre élevé pour la résolution de l'équation de transport sur des maillages hexagonaux.

Development and comparison of high-order finite element bases for solving the transport equation on hexagonal meshes.

Présentée par :

David LABEURTHRE

Direction de thèse :

Romain LE TELLIER

Directeur de Recherche CEA, Université Grenoble Alpes

Directeur de thèse

Ansar CALLOO

CEA

Co-encadrant de thèse

Rapporteurs :

Jean RAGUSA

PROFESSEUR, Texas A&M University

Bruno DESPRES

PROFESSEUR DES UNIVERSITES, Sorbonne Université

Thèse soutenue publiquement le **8 décembre 2022**, devant le jury composé de :

Romain LE TELLIER

DIRECTEUR DE RECHERCHE, CEA centre de Cadarache

Directeur de thèse

Jean RAGUSA

PROFESSEUR, Texas A&M University

Rapporteur

Bruno DESPRES

PROFESSEUR DES UNIVERSITES, Sorbonne Université

Rapporteur

Albert COHEN

PROFESSEUR DES UNIVERSITES, Sorbonne Université

Président

Pablo RUBIOLLO

PROFESSEUR DES UNIVERSITES, Grenoble INP

Examinateur

Olga MULA

PROFESSEUR ASSOCIE, Technische Universiteit Eindhoven

Examinatrice

Stéphanie SALMON

PROFESSEUR DES UNIVERSITES, Université de Reims

Examinatrice

Invités :

Ansar Calloo

INGENIEUR DOCTEUR, CEA Saclay

Emiliano Masiello

INGENIEUR DOCTEUR, CEA Saclay



Remerciements

Un homme a dit un jour : "Voici le périlleux exercice des remerciements !" et je dois bien reconnaître que je suis du même avis, les remerciements publics n'étant pas mon fort. Toutefois, des mots pour exprimer ma gratitude sont bien nécessaires.

Premièrement, je souhaiterais remercier le CEA pour avoir rendu cette thèse possible ainsi que l'ensemble du personnel du SERMA pour leur accueil.

Un grand merci à Jean Ragusa et Bruno Després pour avoir accepté la lourde tâche d'être rapporteurs. Leurs commentaires ont notamment permis d'éclaircir certains points de ce travail.

Merci à Albert Cohen, Stéphanie Salmon, Pablo Rubiolo, Olga Mula et Emiliano Masiello pour avoir fait partie du jury de cette thèse et pour l'intérêt porté à mes travaux.

Des remerciements tout particuliers vont à Ansar Calloo et à Romain Le Tellier dont l'encadrement a été formidable. Sans eux, ce travail n'aurait pas été possible. Je suis sincèrement honoré de les avoir rencontrés et d'avoir pu travailler avec eux. J'espère en avoir été digne. Je me rappellerai encore longtemps des nombreuses discussions et des moments passés avec Ansar qui ont rendu cette thèse vraiment agréable.

Je souhaiterais également remercier l'ensemble des professeurs qui m'ont transmis un profond intérêt pour les sciences et plus spécifiquement pour les mathématiques. Je pense notamment à Dominique Jamin, Eric Reynaud, Philippe Lauron, Gilles Leborgne et Laurent Chupin pour la grande qualité de leurs enseignements.

Des remerciements vont évidemment aux thésards qui ont contribué à rendre le CEA un endroit plaisant, en particulier Emeline, Matteo, Ali, Rodrigue, Yonah avec qui j'ai parfois eu de très (trop) longues discussions.

Merci à mes amis de Paris, de Reims (oui Pierre c'est bien de toi dont je parle) et du sud de la France pour les moments passés ensemble, que ce soit autour d'un verre, d'une partie de pétanque ou durant des échanges musicaux.

Je remercie grandement ma compagne pour m'avoir toujours soutenu et pour avoir fait preuve d'une profonde écoute. Je suis plus que reconnaissant pour son aide.

Merci infiniment à ma famille pour m'avoir supporté tout au long de ma scolarité, y compris durant ces trois années de thèse. Leur soutien est toujours fondamental.

Résumé

En simulation multiphysique des accidents graves des réacteurs à neutrons rapides refroidis au sodium, les temps de calcul sont souvent dominés par la résolution de l'équation de transport pour la neutronique, en particulier pour la phase dite "primaire" de l'accident où la géométrie du cœur composée d'assemblages hexagonaux reste intègre.

Afin de réduire le coût de la neutronique tout en conservant une certaine précision, ce travail propose de résoudre l'équation de transport discrétisée sur des maillages hexagonaux par une méthode de Galerkin discontinue sans sous-maillage des hexagones. À cette fin, des bases de monômes orthogonalisées ainsi que des bases nodales de fonctions rationnelles de Wachspress d'ordre élevé ont été développées pour l'hexagone régulier. En particulier, pour la construction de ces dernières, un algorithme original basé sur l'utilisation du calcul symbolique a été proposé.

Pour traiter des géométries 3D prismatiques, des bases d'éléments finis ont ensuite été développées à partir de ces bases 2D sur l'hexagone et d'une base 1D polynomiale. Au-delà du produit tensoriel des bases 2D et 1D, à un ordre donné k , une combinaison originale de ces bases a été proposée afin d'engendrer l'espace des polynômes de degrés k en trois dimensions avec le moins de degrés de liberté possible.

L'ensemble des bases développées a fait l'objet d'une vérification par la méthode des solutions manufacturées vis-à-vis des ordres de convergence théoriques attendus.

Finalement, des calculs sur des benchmarks de cœurs en deux et trois dimensions ont été réalisés pour comparer les bases d'éléments finis développées sur l'hexagone à des solutions issues des solveurs APOLLO3[®]/MINARET et SNATCH où un sous maillage (en triangles ou losanges) est utilisé. Les résultats montrent que les bases proposées permettent d'améliorer significativement le compromis entre précision et coût de calcul pour le niveau de précision visée dans des applications de couplage multiphysique (typiquement, 10 pcm sur le taux de multiplication effectif).

Mots-clés : l'équation de transport de neutrons, éléments finis polygonaux, bases d'éléments finis d'ordre élevé, bases de Wachspress

Abstract

In the realm of multiphysics simulation of severe accidents for sodium-cooled fast reactors, the computation time is often dominated by the resolution of the neutron transport equation, in particular for the so-called "primary" phase of the accident where the core geometry composed of hexagonal assemblies remains intact.

In order to reduce the cost of the neutronics part while keeping a certain accuracy, this work proposes to solve the transport equation discretized on hexagonal meshes by a discontinuous Galerkin method without submeshing the hexagons. To this end, bases of orthogonalized monomials as well as nodal bases of high-order Wachspress rational functions have been developed for the regular hexagon. In particular, for the construction of the latter, an original algorithm based on the use of symbolic computation has been proposed.

To deal with 3D prismatic geometries, finite element bases were then developed from these 2D bases on the hexagon combined with 1D polynomial bases. Beyond the tensor product of the 2D and 1D bases, at a given order k , an original combination of these bases has been proposed in order to generate the space of polynomials of degree k in three dimensions with the least number of degrees of freedom possible.

These bases generated have been verified by the method of manufactured solutions with respect to the expected theoretical convergence rates.

Finally, calculations on benchmarks of two and three dimensional cores have been performed to compare the finite element bases developed on the hexagon with solutions from the solvers APOLLO3[®]/MINARET and SNATCH where a sub-mesh (in triangles or diamonds) is used. The results show that the proposed bases allow to significantly improve the trade-off between accuracy and computational cost for the level of accuracy targeted in multiphysics coupling applications (typically, 10 pcm on the effective multiplication rate).

Keywords : the neutron transport equation, polygonal finite element bases, high-order finite element bases, Wachspress bases

Table des matières

Table des figures	xi
Liste des tableaux	xv
Introduction	1
1 Contexte	11
1 Rappels de neutronique	12
1.1 L'équation de transport des neutrons	12
1.2 Discrétisation énergétique	16
1.3 Modélisation de la section efficace de diffusion	18
1.4 Discrétisation angulaire	20
1.5 Discrétisation de la variable spatiale par une méthode de Galerkin discontinue	22
1.6 Algorithme de résolution de l'équation de transport	29
2 Exemples de bases d'éléments finis pour des polygones	32
2.1 Quelques exemples de coordonnées barycentriques généralisées	32
3 Synthèse	37
2 Développement de bases d'éléments finis sur des polygones convexes	39
1 Forme générale des fonctions rationnelles de Wachspress	40

1.1	Des premiers travaux réalisés par Gout	41
1.2	Extension des travaux de Gout à des ordres élevés	44
2	Une méthode de génération de bases de fonctions rationnelles	47
2.1	Principe général	47
2.2	Remarques sur le système linéaire	52
2.3	Transformation homothétique de la solution obtenue	54
2.4	Détails d'implémentation	55
3	Applications de l'algorithme de construction de base	57
3.1	Application à un hexagone régulier	57
3.2	Application au pentagone irrégulier	61
4	Développement de la base de monômes orthogonalisée	64
5	Conclusion et perspectives	67
3	Résolution d'équations aux dérivées partielles sur des maillages hexagonaux	69
1	Intégration numérique sur l'hexagone régulier	70
1.1	Partitionnement et transformation de Duffy	70
1.2	Intégration de fonctions sur le bord des hexagones	74
1.3	Vérifications numériques	74
2	Vérification des ordres de convergence et premières comparaisons	77
2.1	Principe général	77
2.2	Premières vérifications avec l'équation de Poisson	79
2.3	Une deuxième vérification et des premières comparaisons pour la résolution de l'équation de transport	83
3	Résultats numériques sur deux benchmarks numériques	94
3.1	Application à un benchmark monogroupe	95
3.2	Application à un benchmark à quatre groupes d'énergie	99
4	Conclusion et perspectives	103

4	Extension des bases hexagonales 2D à la 3D	105
1	Bases d'éléments finis 3D à l'aide du produit tensoriel	106
2	Une composition originale de bases d'éléments finis	108
2.1	Une base polynomiale hiérarchique	108
2.2	Génération d'une base d'éléments finis 3D à partir d'une somme de produits tensoriels entre W_k et \tilde{V}_{n-k}	108
3	Synthèse	113
5	Résolution de l'équation de transport en 3D	115
1	Vérification des bases d'éléments finis 3D avec les MMS	116
1.1	Solutions dans l'espace d'approximation	116
1.2	Solution $\psi_m \in C^\infty(K), \psi_m \notin P$	117
2	Résultats numériques sur un cas réacteur	119
2.1	Description du benchmark	119
2.2	Résultats numériques	120
3	Conclusion et perspectives	126
	Conclusions et perspectives	127
	Bibliographie	133
	Annexe A Preuves sur les éléments finis nodaux polygonaux	141
1	Preuve que les polynômes ne peuvent être utilisés pour générer des éléments finis de Lagrange	141
2	Preuve de la nécessité d'utiliser des points intérieurs à un ordre $k \geq m$	142
3	Propriétés des fonctions de Wachspress sur le bord d'un polygone	143
	Annexe B Définitions des erreurs associées aux propriétés que doivent vérifier les bases	145

1	Erreur sur les contraintes géométriques	145
2	Erreur de projection	145
Annexe C Intégration spatiale sur l'hexagone		147
Annexe D Espace V_l ainsi que construction des matrices pour les bases multiplicatives		149
1	Espace V_l pour la base multiplicative	149
2	Calcul des différentes matrices avec la base multiplicative	151
2.1	Calcul des matrices de volume	151
2.2	Calcul des matrices de surface	153
Annexe E Résultats Takeda 3D		157

Table des figures

1	Photo tirée de [48] représentant un cœur de RNR-Na, en l'occurrence le réacteur indien PFBR (Prototype Fast Breeder Reactor) en construction à Kalpakkam. Les hexagones correspondent ici aux assemblages.	2
1.1	Angle solide $\vec{\Omega}$	19
1.2	Illustration des notations utilisées	24
1.3	Illustration du balayage cellule par cellule pour une direction $\vec{\Omega}_n$ donnée.	30
1.4	Notations pour les fonctions Mean value	33
1.5	Un quadrilatère quelconque	35
2.1	Notations introduites par Gout dans [46] pour décrire les éléments géométriques nécessaires pour définir des fonctions de Wachspress d'ordre 3	41
2.2	Exemple de fonctions de Wachspress à l'ordre 3.	43
2.3	Éléments géométriques pour définir les fonctions de Wachspress sur un hexagone régulier à l'ordre 3.	45
2.4	Extrait du code Python qui a été utilisé pour produire les fonctions de base.	56
2.5	Illustrations des fonctions de base w_1, w_{1_1} à l'ordre 4. Pour les nœuds d'ancrage a_1, a_{1_1} respectivement, ces fonctions respectent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.	58
2.6	Illustrations des fonctions de base w_{1_2}, w_{1_3} à l'ordre 4. Pour les nœuds d'ancrage a_{1_2}, a_{1_3} respectivement, ces fonctions présentent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.	58
2.7	Le pentagone irrégulier considéré pour notre étude.	61

2.8	Pentagone irrégulier considéré dans ce travail. Les points a_i représentent les sommets du polygone. Les points b_i représentent les EIP qui permettent de définir l'adjoint q du pentagone irrégulier représentée ici par une ellipse en rouge.	62
2.9	Illustrations des fonctions de base à l'ordre 4 : w_3, w_{3_1} pour les noeuds d'ancrage à a_3, a_{3_1} respectivement. Ces fonctions présentent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.	63
2.10	Illustrations des fonctions de base à l'ordre 4 : w_{3_2}, w_{3_3} pour les noeuds d'ancrage à a_{3_2}, a_{3_3} respectivement. Ces fonctions présentent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.	64
3.1	Transformation \mathcal{T} de triangle de référence vers le triangle physique du système.	70
3.2	Transformation du triangle de référence vers le carré unitaire de centre O	72
3.3	Numérotation des arêtes.	74
3.4	Exemple d'un domaine cartésien \mathcal{D} maillé avec des hexagones. Les lignes pointillées correspondent aux bords des cellules fantômes.	79
3.5	Courbes de l'erreur e_h pour différents ordres	83
3.6	$e_h = \ \psi_m - \psi_h\ _{L^2(K)}$ sur le domaine K aux ordres 1-2-3-4-5 pour les bases de monômes orthogonalisées et de Wachspress. Les droites présentant des \diamond et des \circ correspondent respectivement aux résultats pour POLY et HOW.	85
3.7	Variation de l'erreur $\ \psi_m - \psi_h\ _{L^2(K)}$ selon k pour un nombre de cellules $N_K = 10829$ pour la fonction $\psi_m(x, y) = \sin\left(\frac{2\pi x}{L_x}\right) \sin\left(\frac{2\pi y}{L_y}\right)$ avec POLY (\diamond) et HOW (\circ).	87
3.8	Conditions aux limites imposées sur les bords gauche et inférieur du domaine. La ligne droite orange correspond à la ligne caractéristique d'équation $y = (x + L_x/2)\frac{b}{a} - L_y/2$	88
3.9	Illustrations des domaines non aligné et aligné ainsi que du partitionnement d'un hexagone en six triangles.	89
3.10	Illustrations de l'erreur $\ \psi_m - \psi_h\ _{L^2(K)}$ pour les fonction MMS1-MMS2 avec POLY (droites présentant des \diamond) et de Wachspress (droites présentant des \circ).	91

3.11	Variation de l'erreur $\ \Psi_m - \Psi_h\ _{L^2(K)}$ selon k pour un nombre de cellules $N_K = 13633$ pour la fonction MMS1 avec POLY (\diamond) et HOW (\circ).	92
3.12	Variation de l'erreur $\ \Psi_m - \Psi_h\ _{L^2(K)}$ selon k pour un nombre de cellules $N_K = 13633$ pour la fonction MMS2 avec POLY (\diamond) et HOW (\circ).	92
3.13	Benchmark hexagonal 2D	95
3.14	Erreurs sur les taux d'absorption pour $N_{ddl} = 2286$	98
3.15	Benchmark hexagonal 2D issu de [100]	99
3.16	Erreur sur les taux d'absorption avec HOW à l'ordre 5	102
5.1	$e_h = \ \Psi_m - \Psi_h\ _{L^2(K)}$ pour la base additive et la base multiplicative à l'ordre 3 118	
A.1	Quadrilatère sans côtés parallèles	141
C.1	Illustrations des nœuds des quadratures à différents ordres.	147
D.1	Numérotation des faces du dessous et du dessus pour une maille hexagonale 3D prismatique	151
E.1	Taux d'absorption dans les quatre premiers plans à partir du bas du cœur. . .	157
E.2	Taux d'absorption dans les trois derniers plans à partir du bas du cœur. . . .	158

Liste des tableaux

1.1	Propriétés de certaines fonctions coordonnées barycentriques généralisées. Tableau tiré en partie de [3].	35
2.1	Erreurs associées aux différentes bases calculées à l'ordre 3 pour certains hexagones de pas p	60
2.2	Fonctions présentant des variables indépendantes pour l'hexagone régulier .	60
2.3	Variables indépendantes associées aux fonctions r^{ij} pour l'hexagone régulier	60
2.4	Fonctions présentant des variables indépendantes pour le pentagone irrégulier	63
2.5	Variables indépendantes associées aux fonctions r^{ij} pour le pentagone irrégulier	63
3.1	Transformation géométrique de l'intervalle de référence I_0 vers les arêtes hexagonales physiques ($\xi \in I_0$). La numérotation locale des arêtes est donnée par la Figure 1.2.	74
3.2	Erreurs d'intégration numérique sur une cellule hexagonale.	75
3.3	Erreurs d'intégration numérique pour le bord numéro 1.	76
3.4	Quelques résultats numériques sur le calcul de e_h pour différents u_m avec $N_K = 9$	82
3.5	Erreurs e_h pour différents ψ_m avec les bases de monômes orthogonalisées .	84
3.6	Erreurs e_h pour différents ψ_m avec les bases de Wachspress	85
3.7	Sections efficaces	95
3.8	Comparaison du Δk_{eff}	97
3.9	Comparaison des erreurs sur les taux d'absorption	97

3.10	Comparaison du Δk_{eff}	100
3.11	Comparaison des erreurs sur les taux d'absorption	101
4.1	Nombre de fonctions utilisées pour les différentes bases 3D possibles	112
5.1	Erreurs $\ \psi_m - \psi_h\ _{L^2(K)}$ pour trois fonctions ψ_m . Les erreurs sont calculées avec la base multiplicative $P = \text{Vect}(W_k \otimes V_l)$ avec $(k, l) \in \llbracket 1, 3 \rrbracket^2$	117
5.2	Erreurs $\ \psi_m - \psi_h\ _{L^2(K)}$ obtenues pour trois fonctions ψ_m avec les bases additives.	117
5.3	Répartition des matériaux par plan selon une diagonale du cœur (cf. la Figure 3.15)	120
5.4	Comparaison du Δk_{eff} entre la solution de référence et SNS1 ainsi que les bases additives.	121
5.5	Comparaison du Δk_{eff} pour les bases multiplicatives avec $P = \text{Vect}(W_k \otimes V_l)$ où $(k, l) \in \llbracket 1, 2 \rrbracket \times \llbracket 1, 5 \rrbracket$	121
5.6	Comparaison du Δk_{eff} pour les bases multiplicatives avec $P = \text{Vect}(W_k \otimes V_k)$ où $k \in \llbracket 1, 5 \rrbracket$	121
5.7	Erreurs max et rms sur les taux d'absorption entre la solution de référence et SNS1 ainsi que les bases additives.	122
5.8	Erreurs max et rms sur les taux d'absorption pour les bases multiplicatives avec avec $P = \text{Vect}(W_k \otimes V_l)$ où $(k, l) \in \llbracket 1, 2 \rrbracket \times \llbracket 1, 5 \rrbracket$	122
5.9	Erreurs max et rms sur les taux d'absorption pour les bases multiplicatives avec $P = \text{Vect}(W_k \otimes V_k)$ où $k \in \llbracket 1, 5 \rrbracket$	122
D.1	Blending functions pour l'élément fini de référence 1D	150
D.2	fonctions "Mesh entity level" pour l'élément fini de référence 1D	150

Introduction

Mise en contexte et objectif général

Pour répondre aux besoins industriels grandissants, la simulation numérique joue un rôle clé dans le développement et la compréhension de systèmes physiques complexes. Ces impératifs se retrouvent notamment pour la conception et l'analyse des réacteurs nucléaires. En effet, le comportement d'un réacteur nucléaire met en jeu différentes physiques dont la thermomécanique, la mécanique des fluides, la neutronique. Il est alors important de pouvoir simuler chacune d'entre elles. En particulier pour garantir les exigences de sûreté, tout un pan de recherche est dédié à la simulation des "accidents graves" où, selon différents scénarios fonction du type de réacteurs considérés, un ou plusieurs systèmes de contrôle ou d'extraction de la puissance du réacteur sont supposés inopérants.

Considérons le cas d'un transitoire de type "Unprotected Loss Of Flow" (ULOF) pour les réacteurs à neutrons rapides refroidis au sodium (RNR-Na) qui correspond à la situation où le circuit de refroidissement est défaillant. Pour ces cœurs composés d'assemblages de combustible entourés de boîtiers hexagonaux en acier, le transitoire accidentel est découpé en deux phases :

- la "phase primaire" tant que les boîtiers hexagonaux sont intègres et que la dégradation (échauffement, fusion et déplacement de matériaux initialement solides) reste intra-assemblage *i.e.* le cœur conserve une géométrie hexagonale ;
- la "phase secondaire" où la dégradation se généralise et un bain de matériaux fondus nommé corium se forme dans le cœur.

Une illustration d'assemblages hexagonaux dans un cœur de réacteur est fournie à la Figure 1.

La simulation d'un tel accident nécessite des études paramétriques mettant en jeu un couplage entre la neutronique et la thermohydraulique (incluant la dégradation). La thermohydraulique requiert la puissance qui est déterminée par la neutronique là où cette dernière dépend de la température du combustible et de la répartition spatiale des matériaux. Citons

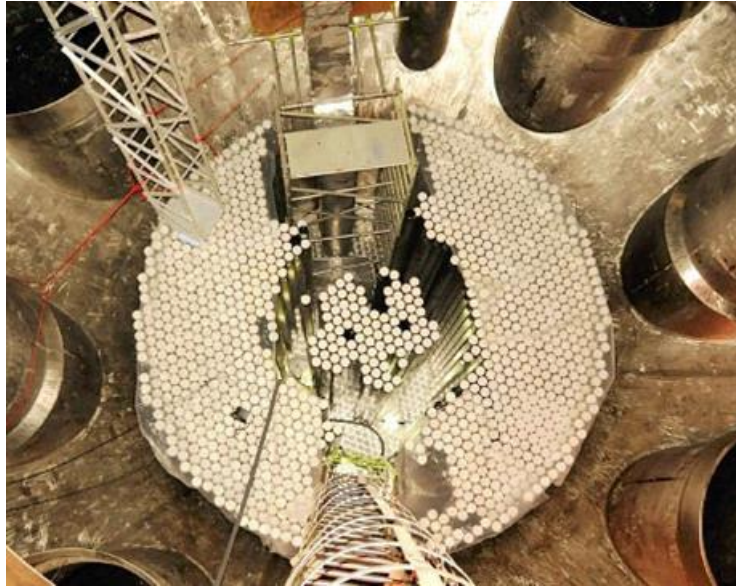


FIGURE 1 Photo tirée de [48] représentant un cœur de RNR-Na, en l'occurrence le réacteur indien PFBR (Prototype Fast Breeder Reactor) en construction à Kalpakkam. Les hexagones correspondent ici aux assemblages.

en particulier le code SIMMER [101] où un tel couplage est mis en œuvre pour simuler ce type de transitoires accidentels.

Dans la phase primaire de l'accident, comme il n'y a pas de transfert de masse entre assemblage, la modélisation thermohydraulique multiphasique peut être simplifiée par une approche "multi-canal" qui s'apparente à une décomposition de domaine. À la limite où l'on néglige les transferts de chaleur inter-assemblages, les différents assemblages sont indépendants du point de vue de leur comportement thermohydraulique. De par les fortes incertitudes sur la phénoménologie de la dégradation intra-assemblage, les modèles de thermohydraulique mis en œuvre ne traitent pas de manière détaillée la géométrie interne de l'assemblage et les grandeurs associées sont essentiellement moyennées dans le plan de l'hexagone.

Par contre, la neutronique, elle, ne peut pas recourir à ce genre d'hypothèse et nécessite un calcul de cœur complet quelque soit la phase de l'accident. En conséquence, au moins dans la phase primaire, la durée de la simulation du transitoire accidentel est dominée par la partie neutronique [49, 50]. Ainsi, la réduction du coût du calcul neutronique dans ce contexte est un axe important d'amélioration de ces simulations multiphysiques.

Bien sûr, la méthode de discrétisation temporelle et le parallélisme sont des axes importants de gain en performances pour cette résolution du problème de neutronique. Pour

autant, un autre axe important et intrinsèque au solveur de l'équation de transport des neutrons mérite d'être exploré. Il touche au schéma de discrétisation spatiale. En effet, pour les réacteurs à géométrie hexagonale, les méthodes de discrétisation spatiale couramment mises en oeuvre en neutronique ne traitent pas directement ces géométries. Par exemple, le solveur APOLLO3[®]/MINARET (voir [63]) propose une solution en sous-maillant à minima un hexagone en six triangles, conduisant alors à un nombre élevé de degrés de liberté. Par conséquent, pour ces discrétisations, les simulations ne sont alors plus réalisables au vu de la durée des temps de calcul de neutronique. Une alternative proposée dans le code SIMMER est d'approcher la géométrie hexagonale en une géométrie cartésienne [73]. Les temps de calculs sont alors plus courts, en revanche, la précision sur le calcul du flux neutronique est amoindrie.

Par conséquent, ces travaux de thèse ont pour objectif de proposer une discrétisation spatiale adaptée aux maillages hexagonaux pour la résolution de l'équation de transport afin d'améliorer le compromis entre précision et temps de calcul dans ce contexte de l'étude des accidents des RNR-Na et, de manière plus générale, dans le contexte de couplage multiphysique.

À cette fin, nous devons résoudre numériquement l'équation de Boltzmann appliquée au transport des neutrons dans le cœur d'un réacteur nucléaire. Cette équation est une équation hyperbolique provenant d'une équation bilan vis-à-vis de la population de neutrons. Sa résolution conduit à l'obtention du flux de neutrons, *i.e.* le produit entre la densité de neutrons (par unité de volume, d'angle solide et d'énergie) et le module de la vitesse à l'énergie considérée. À partir de ce flux, différentes grandeurs neutroniques peuvent être obtenues comme la puissance, les taux de réaction¹.

La résolution numérique de l'équation de transport peut s'opérer à l'aide de méthodes stochastiques ou déterministes (cf. [53, 70, 88] pour plus d'informations). Ici, ces travaux portent sur l'approche déterministe en considérant la forme intégral-différentielle d'ordre un de l'équation de transport. Dans ce cadre, une succession de discrétisations sont effectuées pour les variables énergétique, angulaire et spatiale [53].

Pour le traitement de la variable énergétique, le domaine associé est partitionné en un ensemble de sous-domaines sur lesquels l'équation de transport est intégrée de telle sorte qu'un système discret d'équations couplées est obtenu. Cette méthode s'appelle la méthode multigroupe.

1. Nombre d'interactions entre le flux et les noyaux d'un milieu, par unité de volume, pour une réaction donnée.

La discrétisation de la variable angulaire s'effectue le plus souvent par des méthodes P_n ou S_n . La méthode P_n consiste à développer la variable angulaire sur la base des harmoniques sphériques là où la méthode S_n discrétise, par une formule de quadrature, la sphère unité en un ensemble de directions pour ensuite résoudre l'équation de transport pour chacune d'entre elle. Dans le cadre de la simulation des réacteurs RNR-Na, la méthode S_n est le plus souvent employée (c'est le cas dans le code SIMMER par exemple) et c'est cette méthode que nous considérons ici.

Enfin, pour discrétiser la variable spatiale, il existe une multitude de méthodes possibles dont la méthode des éléments finis. Cette méthode consiste à mailler le domaine spatial à l'aide de polygones (resp. polyèdres) en 2D (resp. 3D)² sur lesquels l'équation de transport est discrétisée et écrite dans une formulation faible. Pour chaque élément, des espaces de fonctions sont définies et servent de base pour le développement du flux neutronique. À partir de ces fonctions, un système matriciel peut être construit et résolu pour en déduire le flux approché. Cette méthode présente différents avantages (cadre mathématique bien posé, précision, adaptabilité), elle a également été intensivement étudiée dans la littérature et appliquée à un grand nombre de problèmes physiques [7, 99, 108]. Cependant, dans notre cas, l'équation de transport doit être résolue pour chaque groupe et pour chaque direction. Par conséquent, l'empreinte mémoire associé à l'assemblage de matrices globales est très grande de telle sorte que, même dans des environnements distribués en mémoire, cette approche n'est que rarement mis en œuvre jusqu'à maintenant. Une alternative est alors d'utiliser la méthode de **Galerkin discontinue** (DG) avec le schéma décentré amont [15]. Celle-ci présente l'avantage de permettre des résolutions locales de l'équation de transport. Nous considérons cette discrétisation pour la variable spatiale dans ce travail.

Dans l'optique d'utiliser cette méthode sur des mailles hexagonales, nous nous intéressons plus particulièrement au développement de bases d'ordre élevé afin de réduire les erreurs d'approximation sans pour autant avoir à redécouper les mailles hexagonales.

Aussi, il est important de souligner que dans les problèmes de neutronique uniquement, les efforts de recherche sont consacrés à la modélisation des calculs directs du cœur entier pour faire de la simulation haute-fidélité. Dans ce cadre, le maillage spatial est très raffiné [69] et donc l'utilisation de schémas d'ordre élevé n'est pas nécessaire. Cependant, les simulations d'accidents graves, de par le couplage multiphysique, mettent en jeu des quantités physiques (par exemple les températures ou les densités) homogénéisées sur l'assemblage hexagonal. Ainsi, le maillage est grossier comparé aux simulations haute-fidélité.

2. Le plus souvent, des triangles ou des quadrilatères en 2D et des tétraèdres ou des hexaèdres en 3D.

Travaux antérieurs

Pour développer des bases d'éléments finis sur des polygones, un des précurseurs a été Wachspress en 1975 dans [103] où il a proposé des **bases nodales d'éléments finis rationnelles** aussi appelées **fonctions de Wachspress**. Tout d'abord, Wachspress a prouvé qu'il ne pouvait pas y avoir de bases nodales polynomiales d'éléments finis sur des polygones et qu'un moyen d'en développer était d'utiliser une classe de fonction appelée **Coordonnées Barycentriques Généralisées** (Generalized Barycentric Coordinates : GBC). Il a démontré que ces fonctions ne sont plus uniques pour des polygones avec $m \geq 4$ côtés³. Également, il a prouvé que ces fonctions rationnelles pouvaient être utilisées pour développer des éléments finis sur des polygones convexes. Cela lui a permis de résoudre des **équations aux dérivées partielles** (EDP) sur des maillages polygonaux. Cependant, deux principales limitations ont freiné l'essor de ces méthodes. La première provient de la complexité du calcul des fonctions de bases car elles nécessitent le calcul de l'adjoint du polygone. La deuxième limite est la difficulté de calculer ces fonctions à un ordre supérieur à 1.

Gout a poursuivi les travaux de Wachspress en développant des bases d'éléments finis pour plusieurs polygones convexes [46] à des ordres supérieurs ou égaux à 1. En particulier, Gout a proposé des bases sur l'hexagone régulier jusqu'à l'ordre 3. Cependant, aucune méthode n'a été fournie pour calculer ces bases à des ordres supérieurs.

Concernant la résolution de l'équation de transport sur des maillages polygonaux, on peut notamment citer les travaux de Davidson [30], de Hackemack et de Ragusa [51]. Davidson a été le premier à publier des résultats sur la résolution de l'équation de transport sur des quadrilatères avec des fonctions de Wachspress. Toutefois, ses travaux se limitent à des fonctions d'ordre 1. Dans [51], Hackemack et Ragusa ont utilisé différentes GBC dont les fonctions de Wachspress pour résoudre l'équation de transport sur des maillages polygonaux. Ils ont également appliqué la méthode proposée par Gillette dans [83] pour développer des bases d'éléments finis à l'ordre 2 par produit de GBC. Ces travaux ont permis de comparer les erreurs d'approximation entre le flux analytique et le flux approché à partir de **la méthode des solutions manufacturées** (The method of manufactured solutions : MMS). Ces erreurs ont été calculées pour des polygones quelconques dont l'hexagone et pour différentes fonctions de base. En régime asymptotique, ces résultats ont montré des ordres de convergence numériques similaires aux ordres de convergence théoriques attendus. Cependant, ces études se limitent aux ordres 1-2. Aussi, il n'y a pas de proposition pour monter à des ordres supérieurs.

3. Il est important de noter que le triangle et le parallélogramme sont des cas particuliers. Pour ces deux polygones, les coordonnées barycentriques sont polynomiales. De plus, dans le cas du triangle, ses coordonnées barycentriques sont uniques.

Des travaux récents ont été réalisés en parallèle des nôtres par Arbogast et Wang en 2022 [6] sur le développement de bases de Sérendip à un ordre quelconque et sur des éléments finis mixtes pour des polygones convexes. Ces bases présentent l'avantage d'avoir le moins de degrés de liberté possible. Actuellement, ces bases ont seulement été utilisées comme bases d'éléments finis pour la résolution de l'équation de Poisson. Cependant, trois types de maillage ont été utilisés. Un premier maillage constitué de quadrilatères, un deuxième avec des pentagones et enfin, un maillage avec des hexagones. À nouveau, en régime asymptotique, les ordres de convergences numériques obtenus coïncident avec les ordres de convergence théoriques.

Si les bases présentées auparavant sont des bases nodales, dans le cadre de la méthode de Galerkin discontinue, d'autres types de bases sont possibles. En effet, cette méthode n'impose pas la continuité aux interfaces permettant alors l'utilisation de bases **hybrides**, *i.e.* **modales-nodales**, soit purement **modales**. Par exemple, les bases de Legendre ou les bases de monômes orthogonalisées constituent des bases modales. Ces fonctions présentent l'avantage de pouvoir augmenter l'ordre de la base facilement. En conséquence, un nombre important de travaux ont été réalisés avec ces bases d'éléments finis d'ordre élevé sur des polygones. Par exemple, en mécanique des fluides avec Vu-Huu dans [102], en électromagnétisme avec Sovinec dans [96], ou encore en mécanique du solide avec Rank dans [84]. En ce qui concerne la résolution de l'équation de transport, le solveur APOLLO3[®]/NYMO [12] utilise une discrétisation P_n avec développement sur une base d'harmoniques sphériques pour les fonctions angulaires et une méthode DG en espace avec des bases de monômes orthogonalisées sur des maillages non structurés et à bords courbes.

Soulignons qu'il n'y a que peu de comparaisons entre des bases nodales et modales pour des polygones. Pour ce qui concerne l'équation de transport neutronique, l'article [30] présente une comparaison entre des fonctions de Wachspress et la base de monômes à l'ordre 1 sur des quadrilatères. Cette comparaison s'effectue au travers de la méthode des solutions manufacturées. De manière générale, les erreurs entre la solution analytique et la solution approchée sont inférieures avec la base rationnelle qu'avec la base de monômes.

Ces premiers éléments de littérature conduisent à plusieurs approches possibles afin de développer un solveur de transport sur des maillages hexagonaux. Un premier point de départ est d'étendre les travaux de Gout pour développer des bases d'éléments finis de Wachspress à des ordres supérieurs. Il est également intéressant, au vu de notre discrétisation spatiale par une méthode de Galerkin discontinue, de considérer des bases modales avec en particulier celles des monômes orthogonalisées. Ces bases ont déjà servi pour la résolution de l'équation de transport et présentent l'avantage d'avoir moins de degré de liberté que les fonctions de

Wachspress. Dans une logique d'évaluer le compromis entre les performances et la précision, il est alors utile de comparer les bases de Wachspress et les bases de monômes.

C'est dans ce cadre que cette thèse a pour but de développer différentes bases d'éléments finis d'ordre élevé sur l'hexagone et de les comparer. Les comparaisons s'effectuent au travers d'erreurs d'approximation entre la solution analytique et la solution approchée en régime asymptotique avec la méthode des solutions manufacturées et via des calculs de benchmarks de neutronique sur des cœurs de réacteurs simplifiés. Pour cela, un solveur pour l'équation de transport sur des maillages hexagonaux a été développé ainsi que différentes bases en deux et trois dimensions. On notera en particulier le développement d'un algorithme permettant la génération de bases d'éléments finis rationnelles d'ordre élevé pour des polygones convexes.

Plan détaillé du document

Ce document se compose en cinq chapitres suivis d'une conclusion et de perspectives.

Le chapitre 1 précise les notions de neutronique et de résolution numérique de l'équation de transport associée mises en œuvre dans cette thèse. Ce chapitre présente aussi quelques exemples de GBC pour développer des bases nodales d'éléments finis sur des polygones convexes.

Le chapitre 2 porte sur la démarche mise en œuvre pour développer des bases d'éléments finis d'ordre élevé sur l'hexagone. Une partie conséquente est dédiée à un algorithme permettant la génération de bases de fonctions de Wachspress d'ordre élevé sur des polygones convexes. L'algorithme se résume en la construction (par calcul symbolique) et la résolution numérique d'un système linéaire à partir des propriétés que doivent vérifier les fonctions de base. Les inconnues de ce système étant les coefficients associés aux fonctions de Wachspress. Cet algorithme a pu être appliqué pour calculer des bases aux ordres 3-4-5 pour l'hexagone régulier et aux ordres 3-4 pour un pentagone irrégulier. Le cas du pentagone irrégulier est utilisé pour démontrer la généralité de la méthode de génération de bases. Ces travaux ont conduit à l'écriture de l'article "*High-order Wachspress functions on convex polygons through computer algebra*" publié dans "Journal of Computational physics" [62]. En parallèle, des bases de monômes orthogonalisées sont présentées et développées. Ces bases présentent l'avantage de pouvoir être générées à un ordre arbitraire.

Le chapitre 3 traite, quant à lui, de la résolution d'équations aux dérivées partielles sur des maillages hexagonaux à partir des bases développées dans le chapitre 2. Ce chapitre se

divise en deux grandes parties : une partie de vérification et une deuxième de comparaison vis-à-vis de l'équation de transport en neutronique. La première s'articule autour de la méthode des solutions manufacturées. L'idée est de vérifier que les ordres de convergence numériques coïncident avec les ordres théoriques en régime asymptotique dans le cas de fonction régulières. En premier lieu, l'équation de Poisson est uniquement résolue avec les fonctions de Wachspress par une méthode des éléments finis continus. Celle-ci sert de cas introductif avant de traiter l'équation de transport dont la discrétisation est plus complexe. Pour cette dernière, les calculs s'effectuent avec les bases de Wachspress et les bases de monômes orthogonalisées. Une fois les ordres de convergence vérifiés, la deuxième étape de comparaison s'ouvre⁴, la démarche consiste à utiliser une nouvelle fois la méthode des solutions manufacturées avec des solutions dont les régularités sont similaires à celles des solutions de l'équation de transport. Les erreurs numériques entre les solutions analytiques et les solutions approchées sont alors étudiées entre les bases de Wachspress et les bases de monômes orthogonalisées. Pour poursuivre les comparaisons et répondre à l'objectif initial de cette thèse, l'équation de transport est résolue sur des cas de benchmarks de cœurs de réacteurs 2D hexagonaux. Plus particulièrement, le benchmark de Hébert [16] et le benchmark de Takeda où nous avons pu comparer nos résultats à ceux des solveurs APOLLO3[®], SNATCH. Les résultats rapportés dans ce chapitre ont conduit à l'écriture de deux articles en conférences internationales ([16, 61]).

Au chapitre 4, à partir des bases développées en deux dimensions, deux méthodes de construction de bases d'éléments finis pour des maillages prismatiques en trois dimensions sont présentées. Une première se base sur le produit tensoriel entre la base radiale 2D et une base axiale 1D, cette base est nommée "base multiplicative". Ce choix existe déjà dans la littérature et est une option disponible par le solveur APOLLO3[®]/MINARET. Pour des bases 2D et 1D d'ordre k , cette approche engendre l'espace des polynômes de degré partiel d'ordre $\leq k$ et non l'espace des polynômes de degré total d'ordre $\leq k$. En neutronique, il est constaté que cette approche n'offre pas forcément le meilleur compromis entre précision et coût de calcul. En particulier, en pratique, les utilisateurs de APOLLO3[®]/MINARET favorisent l'option additive de ce dernier qui correspond à une base 3D partielle construite par la somme d'une base radiale d'ordre k et d'une base axiale d'ordre l . Cette approche nous a conduit à développer une méthode de construction de bases 3D originale que nous avons également nommée "base additive". Cette dernière présente l'avantage d'avoir beaucoup moins de degrés de liberté que la base multiplicative tout en engendrant l'espace des polynômes d'ordre $\leq k$. Ces travaux ont été présentés à la conférence *27th International Conference on Transport Theory - ICTT27*.

4. Bien que les résultats sur la fonction régulière aient donné lieu à des premiers éléments de comparaison.

Le chapitre 5 présentent des résultats relatifs à la résolution de l'équation de transport sur des maillages hexagonaux 3D extrudés avec les bases additives et multiplicatives. Une première étape consiste à vérifier l'implémentation des bases avec la méthode des solutions manufacturées pour s'assurer que les ordres de convergence numériques sont proches des ordres de convergence théoriques. Enfin, l'équation de transport est résolue sur des benchmarks 3D hexagonaux, notamment le benchmark de Takeda où nous avons pu comparer nos résultats à ceux du solveur SNATCH . Un article présentant ces résultats et la construction des bases additives est en cours de rédaction.

Chapitre 1

Contexte

Dans l'objectif de comparer des bases d'éléments finis nodales et modales pour la résolution de l'équation de transport sur des hexagones, il est important de présenter les équations mises en jeu ainsi que les bases possibles. Dans ce cadre, ce chapitre commence par des rappels de neutronique présentant les discrétisations énergétique (multigroupe), angulaire (ordonnées discrètes S_n) et spatiale (Galerkin discontinue) de l'équation de transport ainsi que les algorithmes de résolution.

Pour le choix des bases d'éléments finis possibles pour résoudre l'équation de transport sur des hexagones et plus généralement sur des polygones, les fonctions coordonnées barycentriques généralisées sont présentées. Ces dernières constituent des bases nodales d'ordre un. Parmi l'ensemble de ces fonctions, il existe notamment les fonctions de Wachspress qui seront étudiées plus en détail par la suite.

1 Rappels de neutronique

Selon [25], la neutronique est la discipline qui étudie le comportement des neutrons au sein d'un système constitué de noyaux fissiles. Celle-ci fut introduite par l'expérience de Chadwick dans les années 1930 [19] et c'est en 1938 en Allemagne que la fission induite par neutron fut découverte par Otto Hahn, Fritz Strassmann et Lise Meitner. En 1939 en France, Frédéric Joliot-Curie mit en évidence l'émission de deux ou trois neutrons secondaires, lors de la fission. A partir de ces découvertes, les physiciens comprirent que des réactions en chaîne étaient possibles. De ce fait, des études ont été réalisées pour comprendre les conditions d'une réaction en chaîne. En particulier, le facteur de multiplication fut introduit et ce facteur se définit comme le rapport entre l'effectif des neutrons (ou des fissions) à une génération donnée et l'effectif à la génération précédente.

Pour un réacteur nucléaire, c'est dans le cœur (voir Figure 1) qu'a lieu le phénomène physique à l'origine de l'énergie produite. La désintégration artificielle de l'uranium ou du plutonium par fission, sous l'impact d'un neutron, s'accompagne de l'émission de plusieurs neutrons d'énergie moyenne de 2 MeV. Cette émission produit un dégagement de chaleur qui est évacuée par un fluide caloporteur permettant *in fine* le fonctionnement des turbines génératrices d'électricité. Pour contrôler la réaction en chaîne dans un réacteur nucléaire, la population de neutrons est maintenue constante grâce à des dispositifs appropriés (barres de contrôle, poisons consommables). Pour maintenir le réacteur dans un état stable dit *critique*, il est nécessaire de s'assurer que la population de neutrons ne croît pas exponentiellement ni ne décroît jusqu'à ce que le réacteur s'éteigne. Dans ce cadre, la simulation numérique de l'évolution de la population neutronique est alors une étape importante dans la phase de conception et d'étude du fonctionnement d'un réacteur. Ces simulations nécessitent de résoudre numériquement l'équation de transport que nous allons maintenant présenter.

1.1 L'équation de transport des neutrons

L'équation de transport des neutrons ou équation de transport de Boltzmann linéaire est obtenue en établissant un bilan entre les arrivées et les migrations des neutrons dans un élément d'hypervolume ($d^3\vec{r}, dE, d^2\Omega$) autour du point $(\vec{r}, E, \vec{\Omega})$ à un instant t donné, avec \vec{r} la position spatiale dans un domaine \mathcal{D} , E l'énergie et $\vec{\Omega} \in \mathcal{S}^2$ la direction avec \mathcal{S}^2 la sphère unité. Cette équation fait intervenir des grandeurs que l'on appelle données nucléaires avec en particulier les sections efficaces qui correspondent à des probabilités d'interaction neutrons-noyaux caractérisant une réaction spécifique [88]. C'est l'équation du transport

sous forme intégrô-différentielle [53]. Sans perte de généralités vis-à-vis du sujet de cette thèse qui porte sur la discrétisation spatiale de cette équation, nous considérons l'équation de transport stationnaire :

$$\begin{aligned}
& \overbrace{\vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, E, \vec{\Omega})}^{\text{Transport}} + \overbrace{\Sigma_t(\vec{r}, E) \psi(\vec{r}, E, \vec{\Omega})}^{\text{Collision}} \\
& = \overbrace{\int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} d^2\Omega' \Sigma_s(\vec{r}, \vec{\Omega}' \rightarrow \vec{\Omega}, E' \rightarrow E) \psi(\vec{r}, E', \vec{\Omega}')}^{\text{Diffusion}} \\
& + \overbrace{\frac{\chi(E)}{4\pi} \int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} d^2\Omega' \nu \Sigma_f(\vec{r}, E') \psi(\vec{r}, E', \vec{\Omega}')}^{\text{Fission}} \\
& + \underbrace{S_{\text{ext}}(\vec{r}, E, \vec{\Omega})}_{\text{source extérieur}}
\end{aligned} \tag{1.1}$$

où :

- $\psi(\vec{r}, E, \vec{\Omega})$: flux angulaire en un point \vec{r} de l'espace, à une énergie E , et dans une direction $\vec{\Omega}$. Il s'écrit comme le produit de la densité neutronique $n(\vec{r}, \vec{\Omega}, E)$ par la norme de la vitesse $v(E)$
- $\vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, E, \vec{\Omega})$: terme d'advection prenant en compte les particules sortant de l'hypervolume.
- $\Sigma_t(\vec{r}, E, \vec{\Omega}) \psi(\vec{r}, E, \vec{\Omega})$ est le taux de disparition des neutrons de l'hypervolume par interaction avec la matière, avec $\Sigma_t(\vec{r}, E, \vec{\Omega}, t)$ étant la section efficace totale macroscopique.
- $\int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} \Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \rightarrow \vec{\Omega}) \psi(\vec{r}, E', \vec{\Omega}') d^2\Omega'$: source de diffusion, *i.e.* les neutrons diffusent d'une énergie E' vers une énergie E et/ou d'une direction $\vec{\Omega}'$ vers $\vec{\Omega}$. $\Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \rightarrow \vec{\Omega})$ est la section efficace de diffusion qui modélise les interactions du neutron avec la matière par diffusion élastique et inélastique.
- $\frac{\chi(E)}{4\pi} \int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} d^2\Omega' \nu(\vec{r}, E') \Sigma_f(\vec{r}, E') \psi(\vec{r}, E', \vec{\Omega}')$: terme de fission. ν le nombre moyen des neutrons produit par fission et $\chi(E)$ le spectre de fission, il définit la densité des neutrons d'énergie E produit par une fission.
 $\Sigma_f(\vec{r}, E')$ correspond à la section efficace de fission.

Les dépendances angulaires des sections efficaces peuvent être simplifiées si l'on fait l'hypothèse que les neutrons évoluent dans des milieux isotropes :

$$\Sigma_t(\vec{r}, E, \vec{\Omega}) = \Sigma_t(\vec{r}, E) \quad (1.2)$$

$$v\Sigma_f(\vec{r}, E, \vec{\Omega}) = v\Sigma_f(\vec{r}, E) \quad (1.3)$$

$$\Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \rightarrow \vec{\Omega}) = \frac{1}{2\pi} \Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \cdot \vec{\Omega}) \quad (1.4)$$

A partir des hypothèses ci-dessus et en l'absence de source extérieure, l'Equation (1.1) s'écrit de la manière suivante :

$$\begin{aligned} \vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, E, \vec{\Omega}) + \Sigma_t(\vec{r}, E) \psi(\vec{r}, E, \vec{\Omega}) = \\ \frac{1}{2\pi} \int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} \Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \cdot \vec{\Omega}) \psi(\vec{r}, E', \vec{\Omega}') d^2\Omega' \\ + \frac{\chi(\vec{r}, E)}{4\pi} \int_{E_{\min}}^{E_{\max}} dE' v\Sigma_f(\vec{r}, E') \phi_{00}(\vec{r}, E') \end{aligned} \quad (1.5)$$

Avec $\phi_{00}(\vec{r}, E') = \int_{\mathcal{S}^2} d^2\Omega' \psi(\vec{r}, E', \vec{\Omega}')$ le flux scalaire correspondant au moment angulaire d'ordre 0 du flux angulaire projeté sur la base des fonctions harmoniques sphériques réelles $\mathcal{R}_l^m(\vec{\Omega})$. Ces fonctions seront introduites par la suite.

Pour pouvoir obtenir une solution de l'Equation (1.5), celle-ci est complétée avec des conditions aux limites définies sur $\partial\mathcal{D}$ (le bord de \mathcal{D}) et sur \mathcal{S}^2 . En partant du principe que tous les neutrons quittant le domaine \mathcal{D} ne peuvent plus le réintégrer, cela conduit à la condition dite de vide :

$$\psi(\vec{r}, E', \vec{\Omega}) = 0 \quad \vec{r} \in \partial\mathcal{D}, \text{ si } \vec{n} \cdot \vec{\Omega} < 0 \quad (1.6)$$

où \vec{n} est le vecteur normal sortant de la surface $\partial\mathcal{D}$. Dans le cas où la condition ci-dessus est satisfaite, $\partial\mathcal{D}$ est appelée surface entrante et s'écrit sous la forme Γ^- (voir l'Equation (1.7)) tout au long de ce travail.

$$\Gamma^- = \{\vec{r} \in \partial\mathcal{D}, \vec{n} \cdot \vec{\Omega} < 0\} \quad (1.7)$$

Il est important de noter que l'Equation (1.5) est un problème homogène avec pour solution $\psi = 0$ comme il est expliqué dans [55]. La solution $\psi = 0$ est en générale l'unique solution, les solutions non triviales sont obtenues pour des valeurs particulières de section

efficace. Bien que cela semble paradoxal, en réalité, de par le caractère éminemment transitoire de l'évolution de la population de neutrons, le coeur n'atteint pas des conditions suffisamment stables de sorte à ce que le flux ne vérifie pas exactement l'Equation (1.5). Pour éviter de faire des calculs transitoires, l'idée est de considérer une représentation du système dans des conditions presque stables. Ainsi, l'Equation (1.5) est transformée en un problème à valeurs propres de la forme :

$$\left\{ \begin{array}{l} \text{Trouver } (\zeta, \psi) \text{ tel que :} \\ (\mathcal{A} - \mathcal{B})\psi(\vec{r}, E, \vec{\Omega}) = \zeta \mathcal{F}\psi(\vec{r}, E, \vec{\Omega}) \\ \psi(\vec{r}, E, \vec{\Omega}) = 0, \forall \vec{r} \in \Gamma^- \end{array} \right. \quad (1.8)$$

où :

$$\begin{aligned} - \mathcal{A}\psi(\vec{r}, E, \vec{\Omega}) &= \vec{\Omega} \cdot \vec{\nabla}\psi(\vec{r}, E, \vec{\Omega}) + \Sigma_t(\vec{r}, E)\psi(\vec{r}, E, \vec{\Omega}) \\ - \mathcal{B}\psi(\vec{r}, E, \vec{\Omega}) &= \frac{1}{2\pi} \int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} d^2\Omega' \Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \cdot \vec{\Omega}) \psi(\vec{r}, E', \vec{\Omega}') \\ - \mathcal{F}\psi(\vec{r}, E, \vec{\Omega}) &= \frac{\chi(E)}{4\pi} \int_{E_{\min}}^{E_{\max}} dE' \int_{\mathcal{S}^2} d^2\Omega' \nu \Sigma_f(\vec{r}, E') \psi(\vec{r}, E', \vec{\Omega}') \end{aligned}$$

Une telle solution existe si la production de neutrons par fission équilibre exactement l'évacuation des neutrons par absorption ou fuite du domaine. En notant ζ_m la plus grande valeur propre de l'Equation (1.8), la grandeur caractérisant cet équilibre est notée $k_{eff} = \frac{1}{\zeta_m}$ et se nomme "facteur de multiplication effectif de neutrons". Cette grandeur permet de distinguer trois états pour le réacteur. Si $k_{eff} = 1$, alors le réacteur est dans un état dit "critique", il y a autant de neutrons produits que de neutrons "disparus". Si $k_{eff} > 1$, le réacteur est dit "surcritique", ce cas correspond à une production exponentielle de neutrons. Enfin, si $k_{eff} < 1$, la population de neutrons décroît jusqu'à ce que le réacteur s'éteigne.

Mathématiquement, k_{eff} correspond à la plus grande valeur propre de l'équation de transport, et le flux de neutrons associé est appelé mode fondamental. Cette grandeur étant nécessaire pour pouvoir résoudre l'équation de transport, celle-ci est prise en compte dans toutes les équations suivantes.

Pour obtenir une solution approchée de l'équation de transport stationnaire des neutrons, des discrétisations énergétique, angulaire et spatiale sont nécessaires.

1.2 Discrétisation énergétique

La méthode usuelle pour discrétiser la variable énergétique de l'équation de transport est l'approximation multigroupe. Elle correspond à une subdivision de l'espace $[E_{\min}, E_{\max}]$ en intervalles d'énergie discrets, appelés "groupes" : $\mathbb{E} = \bigcup_{g=1}^G \mathbb{E}_g = \bigcup_{g=1}^G]E_g, E_{g-1}]$, où les sections efficaces sont supposées constantes. Par convention, lorsque g augmente, l'énergie décroît. Par conséquent, l'énergie la plus élevée se voit attribuer le plus petit numéro de groupe. Ainsi, l'Equation (1.5) est intégrée sur les groupes d'énergie de G , ce qui conduit à G équations couplées par le terme de diffusion et de source de fission :

$$\begin{aligned} \vec{\Omega} \cdot \vec{\nabla} \psi^g(\vec{r}, \vec{\Omega}) + \Sigma_t^g(\vec{r}) \psi^g(\vec{r}, \vec{\Omega}) &= \frac{1}{2\pi} \sum_{\mathbb{E}_{g'}} \int_{\mathcal{S}^2} \Sigma_s^{g' \rightarrow g}(\vec{r}, \vec{\Omega}' \cdot \vec{\Omega}) \psi^{g'}(\vec{r}, \vec{\Omega}') d^2\Omega' \\ &+ \frac{\chi^g}{4\pi k_{eff}} \sum_{\mathbb{E}_{g'}} \nu \Sigma_f^{g'}(\vec{r}) \phi_{00}^{g'}(\vec{r}) \end{aligned} \quad (1.9)$$

Le flux angulaire de neutron multigroupe s'écrit comme : $\psi^g(\vec{r}, \vec{\Omega}) = \int_{\mathbb{E}_g} dE' \psi^g(\vec{r}, E', \vec{\Omega})$. Ce flux multigroupe est utilisé comme fonction de pondération pour les termes intervenant de l'équation de transport :

$$\Sigma_t^g(\vec{r}, \vec{\Omega}) = \frac{\int_{\mathbb{E}_g} dE \Sigma_t(\vec{r}, E) \psi(\vec{r}, E, \vec{\Omega})}{\psi^g(\vec{r}, \vec{\Omega})} \quad (1.10)$$

$$\Sigma_s^{g' \rightarrow g}(\vec{r}, \vec{\Omega}' \cdot \vec{\Omega}) = \frac{\int_{\mathbb{E}_{g'}} dE' \int_{\mathbb{E}_g} dE \Sigma_s(\vec{r}, E' \rightarrow E, \vec{\Omega}' \cdot \vec{\Omega}) \psi(\vec{r}, E', \vec{\Omega}')}{\psi^{g'}(\vec{r}, \vec{\Omega})} \quad (1.11)$$

$$\nu \Sigma_f^{g'}(\vec{r}) = \frac{\int_{\mathbb{E}_{g'}} dE' \nu \Sigma_f(\vec{r}, E') \phi_{00}(\vec{r}, E')}{\phi_{00}^{g'}(\vec{r})} \quad (1.12)$$

$$\chi^g(\vec{r}) = \int_{\mathbb{E}_g} dE \chi(\vec{r}, E) \quad (1.13)$$

Cependant, le flux de neutrons est une inconnue de l'équation de transport. Également, la pondération des sections efficaces par le flux angulaire conduit à une dépendance angulaire

supplémentaire de certaines sections ce qui complexifie les équations multigroupes. Pour résoudre ces problèmes, le flux de pondération est obtenu en approximant le flux de neutrons sous la forme d'un spectre d'énergie approprié et représentatif de la physique du problème considéré. Par exemple, dans le cas d'un réacteur à eau pressurisée (REP), trois domaines d'énergie sont identifiés. Pour chaque domaine d'énergie, un spectre est défini. Par exemple, un spectre de Cranberg est utilisé dans le domaine des hautes énergies [26]. Ce spectre a été déduit en étudiant des résultats expérimentaux sur la fission de l'Uranium 235. Pour la région de ralentissement (de l'ordre de la centaine de keV), le flux est modélisé par une fonction $1/E$; enfin, pour le domaine thermique (de l'ordre de la dizaine de meV), le flux est modélisé par une distribution de Maxwell, cf. [72].

Par ailleurs, cette utilisation du spectre d'énergie n'est valable que dans le cas où la variation en énergie des données nucléaires est régulière. Dans le cas de domaines présentant des résonances *i.e.* où les sections efficaces varient de manière significative, cette approximation n'est plus valide. Ainsi, les sections efficaces par groupe doivent être corrigées et cette étape de "correction" est appelée autoprotection [24].

Un point clé réside dans l'"impossibilité" (prise au sens temps de calcul de temps très longs) de simuler l'équation de transport par des méthodes déterministes pour réaliser des calculs de cœur direct. Dans ce cadre, un schéma de calcul à deux étapes procédant par homogénéisation est utilisé [92].

- Étape 1 : Calcul réseau. Il s'agit de calculer précisément en espace et en énergie un motif élémentaire du cœur (assemblage combustible, barre de contrôle, etc.) afin d'obtenir le flux neutronique et les différents taux de réactions sur ce domaine discrétisé spatialement et énergétiquement. Pour ces différents motifs, des propriétés moyennes en espace et en énergie sont ensuite obtenues à l'aide d'une homogénéisation. On notera en particulier une condensation énergétique pour obtenir des propriétés moyennes "équivalentes" sur un maillage énergétique plus grossier. Dans ce cadre, l'autoprotection est une sous-étape du calcul réseau et permet de réduire le nombre de groupes de la bibliothèque multigroupe d'entrée.
- Étape 2 : Calcul du cœur. L'objectif est de résoudre l'équation de transport des neutrons sur l'ensemble du cœur en utilisant les données nucléaires issues du calcul réseau.

D'un point de vue mathématique, ces deux étapes correspondent à un problème multi-échelle. En effet, la première étape correspond à un calcul précis sur plusieurs sous-ensembles

du cœur alors que la deuxième étape correspond à un calcul “grossier” sur l’ensemble du cœur. Pour lier ces deux étapes, une homogénéisation est réalisée à la fin de la première étape. Ainsi, les données générées peuvent être utilisées pour la seconde étape pour résoudre l’équation de transport sur un cœur complet.

1.3 Modélisation de la section efficace de diffusion

Avant de discrétiser la variable angulaire, une première approximation est réalisée en amont sur la section efficace de diffusion multigroupe $\Sigma_s^{g' \rightarrow g}(\vec{r}, \vec{\Omega}' \cdot \vec{\Omega})$. Dans ce travail, nous utilisons la méthode la plus courante qui consiste à développer les sections efficaces de diffusion sur une série tronquée de polynômes de Legendre [75]. Ces polynômes forment une base complète sur $[-1, +1]$:

$$\Sigma_s^{g' \rightarrow g}(\vec{r}, \vec{\Omega}' \cdot \vec{\Omega}) \approx \sum_{l=0}^L \frac{2l+1}{2} \Sigma_{sl}^{g' \rightarrow g}(\vec{r}) P_l(\vec{\Omega}' \cdot \vec{\Omega}) \quad (1.14)$$

En notant $\tilde{\mu} = \vec{\Omega}' \cdot \vec{\Omega}$, les moments de la section efficace de diffusion se définissent comme :

$$\Sigma_{sl}^{g' \rightarrow g}(\vec{r}) = \int_{-1}^{+1} d\tilde{\mu} \Sigma_s^{g' \rightarrow g}(\vec{r}, \tilde{\mu}) P_l(\tilde{\mu}) \quad (1.15)$$

De plus, en utilisant le théorème d’addition des harmoniques sphériques réelles ($\mathcal{R}_l^m(\vec{\Omega})$ voir Equation (1.18)) [53], les polynômes de Legendre peuvent se développer sous la forme :

$$P_l(\vec{\Omega}' \cdot \vec{\Omega}) = \sum_{m=-l}^{+l} \mathcal{R}_l^m(\vec{\Omega}') \mathcal{R}_l^m(\vec{\Omega}) \quad (1.16)$$

En utilisant Equation (1.14) et Equation (1.16), l’équation de transport multigroupe peut s’écrire :

$$\begin{aligned} \vec{\Omega} \cdot \vec{\nabla} \psi^g(\vec{r}, \vec{\Omega}) + \Sigma_t^g(\vec{r}) \psi^g(\vec{r}, \vec{\Omega}) &= \frac{1}{2\pi} \sum_{l=0}^L \frac{2l+1}{2} \sum_{m=-l}^{+l} \mathcal{R}_l^m(\vec{\Omega}) \sum_{\mathbb{E}'_g} \Sigma_{sl}^{g' \rightarrow g}(\vec{r}) \phi_{lm}^{g'}(\vec{r}) \\ &+ \frac{\chi^g}{4\pi k_{eff}} \sum_{\mathbb{E}'_g} \nu \Sigma_f^{g'}(\vec{r}) \phi_{00}^{g'}(\vec{r}) \end{aligned} \quad (1.17)$$

Où $\phi_{lm}^g(\vec{r}) = \int_{\mathcal{S}^2} d^2\Omega \mathcal{R}_l^m(\vec{\Omega}) \psi(\vec{r}, E', \vec{\Omega})$ sont les moments du flux sur les harmoniques sphériques réelles $\mathcal{R}_l^m(\vec{\Omega})$.

En paramétrant $\vec{\Omega}$ sous la forme suivante (Figure 1.1 où $\mu = \cos \theta$), les fonctions harmoniques sphériques réelles $\mathcal{R}_l^m(\vec{\Omega})$ se définissent selon Equation (1.18) :

$$\vec{\Omega} = \begin{pmatrix} \sin \theta \cos \varphi \\ \sin \theta \sin \varphi \\ \cos \theta \end{pmatrix} = \begin{pmatrix} \sqrt{1-\mu^2} \cos \varphi \\ \sqrt{1-\mu^2} \sin \varphi \\ \mu \end{pmatrix}$$

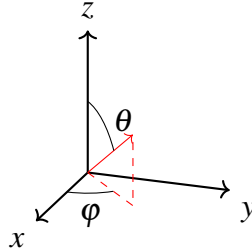


FIGURE 1.1 Angle solide $\vec{\Omega}$.

$$\mathcal{R}_l^m(\vec{\Omega}) = \mathcal{R}_l^m(\mu, \varphi) = \sqrt{(2 - \delta_{m,0}) \frac{(l - |m|)!}{(l + |m|)!}} P_l^{|m|}(\mu) \mathcal{T}_m(\varphi) \quad (1.18)$$

$$\mathcal{T}_m(\varphi) = \begin{cases} \cos(m\varphi) & \text{si } m \geq 0 \\ \sin(|m|\varphi) & \text{si } m < 0 \end{cases} \quad (1.19)$$

$P_l^{|m|}(\mu)$ sont les fonctions de Legendre associées :

$$P_l^{|m|}(\mu) = (1 - \mu^2)^{\frac{|m|}{2}} \frac{d^{|m|}}{d\mu^{|m|}} P_l(\mu). \quad (1.20)$$

Les fonctions P_l sont des polynômes de Legendre qui peuvent se définir selon la formule de Rodrigues :

$$P_l(\mu) = \frac{1}{2^l l!} \left(\frac{d}{d\mu} \right)^{(l)} (\mu^2 - 1)^l \quad (1.21)$$

De cette formule, il peut être déduit :

- $P_0(\mu) = 1$
- $P_1(\mu) = \mu$
- $P_2(\mu) = \frac{(3\mu^2-1)}{2}$
- $P_3(\mu) = \frac{(5\mu^3-3\mu)}{2}$

Quelques remarques peuvent être ajoutées sur les propriétés que vérifient ces polynômes.

Premièrement, ces polynômes sont orthogonaux :

$$\frac{2n+1}{2} \int_{-1}^1 P_n(\mu) \cdot P_m(\mu) d\mu = \delta_{n,m}$$

Deuxièmement, ils vérifient la relation de récurrence :

$$(l+1)P_{l+1}(\mu) - (2l+1)\mu P_l(\mu) + \mu P_{l-1}(\mu) = 0$$

Enfin, ces fonctions forment une base de $L^2([-1, 1])$, ainsi toute fonction $f \in L^2([-1, 1])$ peut se décomposer de la manière suivante :

$$f(\mu) = \sum_{n=0}^{\infty} (2n+1) f_n(\mu) P_n(\mu) \text{ avec } f_n = \frac{1}{2} \int_{-1}^1 f(\mu) P_n(\mu) d\mu$$

1.4 Discrétisation angulaire

Dans ce travail, la variable angulaire $\vec{\Omega}$ est discrétisée en utilisant une méthode de collocation appelée méthode des ordonnées discrètes, ou méthode S_n . Cette méthode a été établie formellement par Chandrasekhar dans le domaine de l'astrophysique pour le transfert radiatif [20], et adaptée à l'équation de transport des neutrons par Carlson [18]. Elle consiste à discrétiser la sphère unité en un ensemble de directions $\vec{\Omega}_n$ associées à des poids ω_n . Ainsi, les intégrales angulaires peuvent se calculer de la manière suivante :

$$\int_{\mathcal{S}^2} d^2\Omega' f(\vec{\Omega}) \approx \sum_{n=1}^N \omega_n f(\vec{\Omega}_n) \quad (1.22)$$

Pour chaque direction dans $\{(w_n, \vec{\Omega}_n)\}_{1 \leq n \leq N}$, l'équation de transport S_n -multigroupe s'écrit :

$$\begin{aligned} \vec{\Omega}_n \cdot \vec{\nabla} \psi^g(\vec{r}, \vec{\Omega}_n) + \Sigma_t^g(\vec{r}) \psi^g(\vec{r}, \vec{\Omega}_n) &= \frac{1}{2\pi} \sum_{l=0}^L \frac{2l+1}{2} \sum_{m=-l}^{+l} \mathcal{R}_l^m(\vec{\Omega}_n) \sum_{\vec{\Omega}'_g} \Sigma_{sl}^{g' \rightarrow g}(\vec{r}) \phi_{lm}^{g'}(\vec{r}) \\ &+ \frac{\chi^g}{4\pi k_{eff}} \sum_{\vec{\Omega}'_g} \nu \Sigma_f^{g'}(\vec{r}) \phi_{00}^{g'}(\vec{r}) \end{aligned} \quad (1.23)$$

où les moments du flux angulaires se calculent ainsi :

$$\phi_{lm}^{g'}(\vec{r}) = \int_{\mathcal{S}^2} d^2\Omega' \mathcal{R}_l^m(\vec{\Omega}') \psi^g(\vec{r}, \vec{\Omega}') = \sum_{n'=1}^N \omega_{n'} \mathcal{R}_l^m(\vec{\Omega}_{n'}) \psi^g(\vec{r}, \vec{\Omega}_{n'}) \quad (1.24)$$

L'équation de transport pour un groupe g donné devient alors un système à N équations, couplée par l'intermédiaire des moments angulaires.

Plusieurs quadratures numériques ont été développées au fil des ans [1] et des recherches se poursuivent [10]. Nous pouvons citer en particulier trois types de quadrature :

- Level-symmetric : la sphère unitaire est divisée en octants, et l'ensemble de quadrature est défini en appliquant des conditions de symétrie de rotation et des poids positifs. Ils sont particulièrement adaptés aux géométries cartésiennes.
- Quadrature produit : les angles polaire et azimutal, respectivement μ et φ sont partitionnés chacun à l'aide d'une règle de quadrature particulière, et la quadrature angulaire finale est le produit de ces deux quadratures. Souvent, μ est développé à l'aide d'une quadrature Gauss-Legendre sur $[0, 1]$ et φ à l'aide d'une quadrature Gauss-Chebyshev sur l'intervalle $[0, \pi/2]$ pour les géométries cartésiennes.
- Quadrature triangulaire : de la même manière qu'une quadrature produit, ces quadratures s'obtiennent à partir d'une quadrature pour μ et d'une quadrature pour φ ; par contre, l'ordre de la quadrature φ varie selon les différentes valeurs de μ de manière à éviter la concentration des points sur les pôles de la sphère que présentent les quadratures produits [104].

Une autre méthode courante pour la discrétisation angulaire est la méthode P_n [53, 70]. Celle-ci consiste à discrétiser la variable angulaire sur la base des harmoniques sphériques. Cependant, la montée en ordre peut conduire à des temps de calcul élevés. Ces travaux de thèse s'effectueront exclusivement avec la méthode S_n . Cette méthode est couramment utilisée pour résoudre numériquement l'équation de transport, c'est par exemple celle qu'utilise le solveur APOLLO3[®]/MINARET [63].

Dans le cas des géométries hexagonales, la quadrature produit est adaptée pour garantir que l'intégration angulaire sur ces géométries est conforme aux symétries de l'hexagone régulier. Ainsi, pour les géométries 2D dans le plan xy , les directions sont définies sur la moitié supérieure de la sphère unitaire où $\mu \geq 0$, et μ est discrétisé par une quadrature Gauss-Legendre sur $[0, 1]$, φ via une quadrature Gauss-Chebyshev sur l'intervalle $[0, \pi/3]$, et le poids associé à chaque direction est le produit des poids des deux quadratures 1D précédentes. Les directions sont en outre triées entre les sextants du cercle unitaire.

1.5 Discrétisation de la variable spatiale par une méthode de Galerkin discontinue

Dans ces travaux, la discrétisation spatiale s'effectue grâce à une méthode de Galerkin discontinue. Historiquement, cette méthode fut introduite par Reed et Hill dans [86] pour résoudre l'équation de transport de neutrons stationnaire sur des maillages triangulaires. Des travaux successifs sur l'analyse de convergence ont découlé de cette méthode, notamment par Lesaint et Raviart dans [67, 68] et par la suite par Johnson et Pitkäranta dans [59]. Ces derniers ont amélioré les résultats sur l'erreur d'estimation trouvés par Lesaint et Raviart en précisant que les ordres de convergence sont de $k + \frac{1}{2}$ en norme L^2 pour une triangulation quelconque dans le cas où une base polynomiale d'ordre k est utilisée pour approximer une fonction régulière. Pour des rectangles [68] ou des maillages triangulaires particuliers [89], l'ordre de convergence est de $k + 1$. Les méthodes DG ont été intensivement étudiées dans la littérature et appliquées pour différentes EDP. En particulier, nous pouvons citer les travaux de Cockburn, Karniadakis et Shu dans [22] pour le développement de méthodes DG pour des équations de convection-diffusion. Les travaux de Ern et Guermond dans [34–36] ont également contribué à l'analyse de méthodes DG pour des systèmes de Friedrichs que ce soit pour des EDP hyperboliques ou elliptiques.

Le principal attrait pour les méthodes DG est l'absence d'hypothèse de continuité entre deux mailles. Ainsi, des modifications locales sont possibles facilitant alors la mise en place de méthodes de raffinement en espace [41]. Aussi, en fonction du choix du schéma numérique utilisé, il est possible de ne pas avoir à assembler une matrice globale. C'est notamment le cas du schéma amont [15, 32] qui conduit à la résolution de problèmes locaux pour le cas de l'équation de transport, *i.e* maille par maille. Dans notre situation, l'équation de transport S_n -multigroupe doit être résolue pour chaque groupe d'énergie et pour chaque direction. Par conséquent, résoudre des problèmes locaux sans la construction d'une matrice globale permet alors la réduction de temps de calcul et est propice au développement de méthodes

de calcul parallèle. Cette démarche est en adéquation avec l'objectif de trouver le meilleur compromis précision/temps de calcul.

Considérons l'équation de transport de neutrons pour une direction $\vec{\Omega}_n$ où l'on note $\psi^g(\vec{r}, \vec{\Omega}_n) = \psi_n^g(\vec{r})$ et pour un groupe d'énergie donné (par souci de lisibilité, l'indice g de groupe est omis dans ce qui suit) :

$$\begin{cases} \vec{\Omega}_n \cdot \vec{\nabla} \psi_n(\vec{r}) + \Sigma_t(\vec{r}) \psi_n(\vec{r}) = \vec{Q}_n(\vec{r}) & \forall \vec{r} \in \mathcal{D} \\ \psi_n(\vec{r}) = \psi_n^{BC}(\vec{r}) & \forall \vec{r} \in \Gamma_- \end{cases} \quad (1.25)$$

avec $\vec{Q}_n(\vec{r})$ le membre de droite de l'Equation (1.23) dans la direction $\vec{\Omega}_n$.

Dans [29, 31], il est prouvé que cette équation admet une unique solution si $Q_n \in L^p(\mathcal{D})$, $\Sigma_t \in L^\infty(\mathcal{D})$, $\psi_n^{BC} \in L^p(\mathcal{D})$, avec $L^p, p \in [1, +\infty]$ les espaces de Lebesgue. Dans ce cadre, la solution ψ appartient à l'espace de Hilbert $H^{\frac{1}{2}-\varepsilon}(\mathcal{D})$ ou $H^{\frac{3}{2}-\varepsilon}(\mathcal{D})$ en fonction des conditions aux limites, comme expliqué dans [29, 59, 60]. Les définitions de ces différents espaces peuvent être trouvées dans [13, 66].

En notant M_h l'ensemble des éléments κ du maillage, introduisons les ensembles suivants :

- $V_h^p = \{v \in L^2(\mathcal{D}) : \forall \kappa \in M_h, v|_\kappa \in Q_p(\kappa)\}$
- $Q_p(\kappa)$: espace des fonctions de base de degré p sur κ

V_h^p correspond à l'ensemble des fonctions générées par $Q_p(\kappa)$ par parties sur chaque κ et discontinues aux interfaces, de degré inférieur ou égal à p .

Afin d'obtenir la formulation faible de l'Equation (1.25), celle-ci est multipliée par une fonction test $\phi_h \in V_h^p$ puis une intégration est réalisée sur un élément κ

$$\int_\kappa \vec{\Omega}_n \cdot \vec{\nabla} \psi_n \cdot \phi_h dV + \int_\kappa \Sigma \psi_n \phi_h dV = \int_\kappa \vec{Q}_n \phi_h dV \quad (1.26)$$

En utilisant le fait que $\vec{\Omega}_n$ est constant, on peut écrire $\vec{\Omega}_n \cdot \vec{\nabla} \psi_n = \text{div}(\psi_n \vec{\Omega}_n)$.

Appliquons maintenant le théorème de Green pour effectuer une intégration par parties :

$$\begin{aligned} \int_\kappa \vec{\Omega}_n \cdot \vec{\nabla} \psi_n \phi_h dV &= \int_\kappa \text{div}(\psi_n \vec{\Omega}_n) \phi_h dV = \\ &- \int_\kappa \vec{\Omega}_n \cdot \psi_n \cdot \vec{\nabla} \phi_h dV + \int_{\partial\kappa} \vec{n} \cdot \vec{\Omega}_n \psi_n \phi_h dS \end{aligned} \quad (1.27)$$

La formule de l'Equation (1.27) permet d'obtenir l'équation suivante :

$$-\int_{\kappa} \vec{\Omega}_n \cdot \vec{\nabla} \phi_h \psi_n dV + \int_{\partial \kappa} \vec{n} \cdot \vec{\Omega}_n \psi_n \phi_h dS + \int_{\kappa} \Sigma \psi_n \phi_h dV = \int_{\kappa} \vec{Q}_n \phi_h dV \quad (1.28)$$

Ici, le bord $\partial \kappa$ est partitionné en deux : une frontière “entrante” et une frontière “sortante” également appelées respectivement “bord éclairé” et “bord non éclairé” :

$$\begin{aligned} \partial \kappa^- &= \left\{ \vec{r} \in \partial \kappa : \vec{\Omega}_n \cdot \vec{n} < 0 \right\} \\ \partial \kappa^+ &= \left\{ \vec{r} \in \partial \kappa : \vec{\Omega}_n \cdot \vec{n} > 0 \right\} \end{aligned} \quad (1.29)$$

Ces notations sont illustrées sur la Figure 1.2.

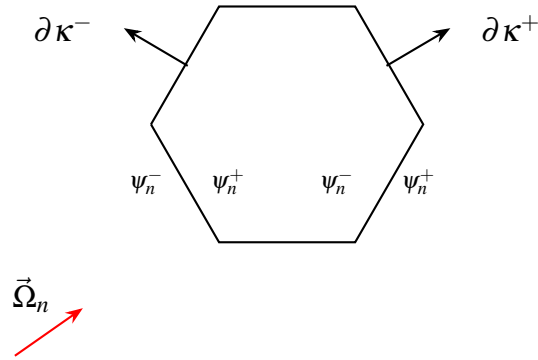


FIGURE 1.2 Illustration des notations utilisées

Ainsi, l'intégration par parties conduit à l'apparition de termes sur le bord de l'élément κ . Or, vu que la discrétisation DG ne garantit pas la continuité entre deux éléments, la définition de traces entrantes et sortantes est nécessaire :

$$\psi_n^\pm(\vec{r} \in \partial \kappa) = \lim_{\varepsilon \rightarrow 0^\pm} \psi_n(\vec{r}_{\text{bord}} + \varepsilon \vec{\Omega}_n).$$

On nomme ψ_n^- le flux amont et ψ_n^+ le flux aval pour la direction angulaire $\vec{\Omega}_n$. Par conséquent, pour un bord donné, il existe deux traces. Pour prendre en compte ces deux composantes, on introduit la notion de *flux numérique* que l'on note $F(\vec{\Omega}_n, \psi_n)$. Pour assurer la stabilité du schéma numérique, nous faisons le choix d'utiliser le schéma décentré amont. Brezzi dans [14] a prouvé que ce schéma décentré amont était équivalent à une pénalisation du saut.

$$\vec{F}(\vec{\Omega}_n, \psi_n) = \begin{cases} \vec{\Omega}_n \psi_n^{BC} & \text{sur } \partial \kappa^- \cap \Gamma \\ \vec{\Omega}_n \psi_h^- & \text{sur } (\partial \kappa^- \setminus \Gamma) \cup (\partial \kappa^+ \cap \Gamma) \end{cases} \quad (1.30)$$

À partir de ce flux numérique, l'Equation (1.28) se réécrit :

$$- \int_{\kappa} \psi_n \vec{\Omega}_n \cdot \vec{\nabla} \phi_h dV + \int_{\partial \kappa} \vec{n} \cdot \vec{F}(\vec{\phi}, \psi_n) \phi_h dS + \int_{\kappa} \Sigma \psi_n \phi_h dV = \int_{\kappa} \vec{Q}_n \phi_h dV \quad (1.31)$$

Décomposons l'intégrale sur $\partial \kappa$:

$$\begin{aligned} \int_{\partial \kappa} \vec{n} \cdot \vec{F}(\vec{\Omega}_n, \psi_n) \phi_h dS &= \int_{\partial \kappa^-} \vec{n} \cdot \vec{F}(\vec{\Omega}_n, \psi_n) \phi_h^+ dS + \int_{\partial \kappa^+} \vec{n} \cdot \vec{F}(\vec{\Omega}_n, \psi_n) \phi_h^- dS \\ &= \int_{\partial \kappa^- \cap \Gamma} \vec{n} \cdot \vec{\Omega}_n \psi_n^{BC} \phi_h^+ dS + \int_{\partial \kappa^- \setminus \Gamma} \vec{n} \cdot \vec{\Omega}_n \psi_h^- \phi_h^+ dS + \int_{\partial \kappa^+} \vec{n} \cdot \vec{\Omega}_n \psi_h^- \phi_h^- dS \end{aligned} \quad (1.32)$$

Comme expliqué dans [41], le flux numérique décentré amont est équivalent à une pénalisation du saut [14] :

$$\vec{n} \cdot \vec{F}(\vec{\Omega}_n, \psi_n) = \frac{\vec{n} \cdot \vec{\Omega}_n}{2} (\psi_n^+ + \psi_n^-) + \frac{|\vec{n} \cdot \vec{\Omega}_n|}{2} (\psi_n^+ - \psi_n^-) \quad (1.33)$$

On obtient alors :

$$\begin{aligned} \int_{\partial \kappa} \vec{n} \cdot \vec{F}(\vec{\Omega}_n, \psi_n) \phi_h dS &= - \int_{\partial \kappa^- \cap \Gamma} |\vec{n} \cdot \vec{\Omega}_n| \psi_h^{BC} \phi_h^+ dS \\ &+ \int_{\partial \kappa^+ \cap \Gamma} |\vec{n} \cdot \vec{\Omega}_n| \psi_h^- \phi_h^- dS + \int_{\partial \kappa^- \setminus \Gamma} |\vec{n} \cdot \vec{\Omega}_n| (\phi_h^- - \phi_h^+) \psi_h^- dS \end{aligned} \quad (1.34)$$

Ainsi, on a la formulation faible suivante :

$$\begin{aligned} - \int_{\kappa} \psi_n \vec{\Omega}_n \cdot \vec{\nabla} \phi_h dV + \int_{\partial \kappa^+ \cap \Gamma} |\vec{n} \cdot \vec{\Omega}_n| \psi_h^- \phi_h^- dS + \int_{\partial \kappa^- \setminus \Gamma} |\vec{n} \cdot \vec{\Omega}_n| (\phi_h^- - \phi_h^+) \psi_h^- dS \\ + \int_{\kappa} \Sigma \psi_n \phi_h dV = \int_{\kappa} \vec{Q}_n \phi_h dV - \int_{\partial \kappa^- \cap \Gamma} \vec{n} \cdot \vec{\Omega}_n \psi_h^{BC} \phi_h^+ dS \end{aligned} \quad (1.35)$$

En pratique, nous utilisons une autre forme faible de l'équation de transport. Pour cela, de la même manière que dans [43], une deuxième intégration par parties est effectuée sur le terme $-\int_{\kappa} \vec{\Omega}_n \cdot \psi_n \cdot \vec{\nabla} \phi_h dV$. La démarche est similaire mais pour la direction $-\vec{\Omega}_n$. Après simplifications, la formulation faible suivante est obtenue :

$$\begin{aligned} & \int_{\kappa} (\vec{\Omega}_n \cdot \vec{\nabla} \psi_n + \Sigma \psi_n) \phi_h dV + \int_{\partial \kappa^- \setminus \Gamma} |\vec{n} \cdot \vec{\Omega}_n| \phi_h^+ (\psi_h^+ - \psi_h^-) dS + \int_{\partial \kappa^- \cap \Gamma} |\vec{n} \cdot \vec{\Omega}_n| \psi_n^+ \phi_h^+ dS \\ &= \int_{\kappa} \vec{Q}_n \phi_h dV + \int_{\partial \kappa^- \cap \Gamma} |\vec{n} \cdot \vec{\Omega}_n| \psi_n^{BC} \phi_h^+ dS \end{aligned} \quad (1.36)$$

Or, en prolongeant la trace sur $\partial \kappa^- \cap \Gamma$, *i.e.* $\psi_n^+ = \psi_n^{BC}$, le problème se simplifie pour obtenir la formulation faible suivante :

$$\begin{aligned} & \int_{\kappa} \left((\vec{\Omega}_n \cdot \vec{\nabla} \psi_n) \phi_h + \Sigma_t \psi_n \phi_h \right) dV - \int_{\partial \kappa^-} (\vec{n} \cdot \vec{\Omega}_n) \psi_n^+ \phi_h^+ dS = \\ & \int_{\kappa} Q_n \phi_h dV - \int_{\partial \kappa^-} (\vec{n} \cdot \vec{\Omega}_n) \psi_n^- \phi_h^+ dS \end{aligned} \quad (1.37)$$

Le problème discret local associé peut donc être formulé de la sorte :

$$\begin{cases} \text{Trouvons } \psi_{n,h} \in V_h^p \text{ tel que, } \forall \phi_h \in V_h^p \\ a_{n,h}(\psi_{n,h}, \phi_h) = l_{n,h}(\phi_h) \end{cases}$$

avec :

$$- a_{n,h}(\psi_{n,h}, \phi_h) = \int_{\kappa} \left((\vec{\Omega}_n \cdot \vec{\nabla} \psi_{n,h}) \phi_h + \Sigma_t \psi_{n,h} \phi_h \right) dV - \int_{\partial \kappa^-} (\vec{n} \cdot \vec{\Omega}_n) \psi_{n,h}^+ \phi_h^+ dS$$

$$- l_{n,h}(\phi_h) = \int_{\kappa} Q_n \phi_h dV - \int_{\partial \kappa^-} (\vec{n} \cdot \vec{\Omega}_n) \psi_{n,h}^- \phi_h^+ dS$$

Pour pouvoir résoudre ce problème, l'espace Q_p doit être défini afin de décomposer ψ_n dans cet espace. Il existe plusieurs choix possibles pour Q_p pour des mailles polygonales, ce point est détaillé dans la section suivante.

Cependant, avant de poursuivre, plusieurs remarques peuvent être faites ici sur la discrétisation spatiale. Bien que le schéma décentré amont assure la stabilité du schéma, dans le cas de l'équation de transport, son intérêt principal réside dans sa propriété à effectuer un balayage pour résoudre des problèmes locaux (*i.e.* maille par maille). Cette caractéristique est importante car, pour rappel, l'équation de transport doit être résolue pour chaque direction

et pour chaque groupe. Dans ce contexte, assembler une matrice globale n'est possible que pour des environnements distribués en mémoire à l'aide de méthodes de décomposition de domaine. Or, les études réalisées ne se déroulent pas forcément dans ce cadre. Également, la stratégie de résolution maille par maille est une propriété importante car en parallélisant celle-ci, des problèmes locaux peuvent être simultanément résolus conduisant alors à une accélération des temps de calculs.

Une limite du schéma décentré amont est le phénomène de locking qui se définit selon Babuška et Suri [8] comme : *“un schéma numérique pour l'approximation d'un problème dépendant d'un paramètre présente un “locking” si la précision des approximations se détériore lorsque le paramètre tend vers une valeur limite. Un schéma numérique est robuste pour le problème s'il converge uniformément pour toutes les valeurs du paramètre”*. Ce phénomène de locking apparaît dans le cas de la limite de diffusion, *i.e.* sous certaines conditions, la solution de l'équation de transport est égale à la solution d'une équation de diffusion. Dans ce cadre, le locking consiste à obtenir une solution de transport différente de la solution de diffusion. Pour éviter ce potentiel problème, Ragusa, Guermond et Kanschat [82] ont adapté le schéma décentré amont en y ajoutant un paramètre d'homothétie dépendant de la paramétrisation des sections efficaces de diffusion.

Aussi, il existe des alternatives au schéma décentré amont pouvant potentiellement offrir *a priori* un meilleur compromis entre précision/nombre de degrés de liberté. C'est notamment le cas des méthodes "Hybrid High Order" (HHO) développées par Di Pietro et Ern [80]. Ces dernières ont été appliquées à différents problèmes physiques dont le transport [31]. Le principe général consiste à reconstruire localement des opérateurs différentiels d'ordre élevé à partir des degrés de liberté basés sur les faces et les cellules d'une maille donnée. Ces degrés de liberté correspondent en réalité à des polynômes d'ordre quelconque. Ces reconstructions locales sont ensuite utilisées pour assembler un système globale. Bien que cette méthode puisse conduire à une précision accrue, cela s'effectue au détriment du caractère local.

Si l'on considère les schémas utilisés en transport permettant une résolution locale et précise, il existe le schéma diamant d'ordre élevé [52]. Toutefois, ces méthodes sont majoritairement utilisées pour des géométries cartésiennes car elles ne sont pas généralisables à n'importe quel polygone convexe.

Problème sous forme matricielle

À partir de la formulation faible (voir l'Equation (1.37)), la formulation matricielle peut alors être déduite, pour cela nous introduisons les fonctions v_h correspondant aux fonctions

de base développées sur l'hexagone (v_h appartient soit à base de monômes, soit à la base de fonctions de Wachspress), la projection de ψ_n dans cette base conduit à :

$$\psi_n(\vec{r}) = \sum_{i=1}^N v_i(\vec{r}) \psi_n^i = \underline{v_\kappa}(\vec{r})^T \underline{\psi_{n,\kappa}} \quad (1.38)$$

où :

- $\underline{v_\kappa}(\vec{r}) = [v_1(\vec{r}), \dots, v_N(\vec{r})]^T$
- $\underline{\psi_{n,\kappa}} = [\psi_n^1, \dots, \psi_n^N]^T$
- $\underline{Q_{n,\kappa}} = [q_n^1, \dots, q_n^N]^T$

Par conséquent, l'Equation (1.37) peut s'écrire :

$$\begin{aligned} & \left(\int_{\kappa} \left(\vec{\Omega}_n \cdot \vec{v_\kappa} \vec{\nabla} v_\kappa^T + \sum_i v_\kappa v_\kappa^T \right) dV \right) \underline{\psi_{n,\kappa}} - \sum_{b \in \partial \kappa_-} \left(\int_b \left(\vec{n}_b \cdot \vec{\Omega}_n \right) v_\kappa v_\kappa^T ds \right) \underline{\psi_{n,\kappa}} \\ &= \left(\int_{\kappa} v_\kappa v_\kappa^T dV \right) \underline{Q_{n,\kappa}} - \sum_{b \in \partial \kappa_-} \left(\int_b \left(\vec{n}_b \cdot \vec{\Omega}_n \right) v_\kappa v_{\kappa^{b,-}}^T ds \right) \underline{\psi_{n,\kappa^{b,-}}} \end{aligned}$$

où la frontière entrante $\partial \kappa_-$ se décompose en un ensemble d'arêtes : b , et $\kappa^{b,-}$ correspond à l'élément adjacent à κ par la frontière b .

Introduisons alors les matrices de volume suivantes :

$$\begin{aligned} \underline{A_\kappa^i} &= \int_{\kappa} v_\kappa \frac{\partial v_\kappa^T}{\partial x_i} dV \\ \underline{M_\kappa} &= \int_{\kappa} v_\kappa v_\kappa^T dV \end{aligned}$$

ainsi que les matrices de surface pour un élément $b \in \partial \kappa_-$,

$$\begin{aligned} \underline{M_{\kappa,b}^{i+}} &= \int_b n_b^i v_\kappa v_\kappa^T ds \\ \underline{M_{\kappa,b}^{i-}} &= \int_b n_b^i v_\kappa v_{\kappa^{b,-}}^T ds \end{aligned}$$

Par conséquent, le système matriciel peut s'écrire de la manière suivante :

$$\left(\sum_i \Omega_n^i \underline{A_\kappa^i} + \sum_{i,\kappa} \underline{M_\kappa} - \sum_{b \in \partial \kappa_-} \sum_i \Omega_n^i \underline{M_{\kappa,b}^{i+}} \right) \underline{\psi_{n,\kappa}} = \underline{M_\kappa} \underline{Q_{n,\kappa}} - \sum_{b \in \partial \kappa_-} \left(\sum_i \Omega_n^i \underline{M_{\kappa,b}^{i-}} \right) \underline{\psi_{n,\kappa^{b,-}}} \quad (1.39)$$

1.6 Algorithme de résolution de l'équation de transport

À partir des discrétisations décrites précédemment, il est possible de résoudre l'équation de transport. Pour cela, l'algorithme de résolution se décompose en trois méthodes itératives imbriquées : les itérations externes, multigroupes et internes [55, 64, 70, 78].

Les itérations externes consistent à mettre à jour les sources de fission pour assurer la convergence sur le k_{eff} . Celle-ci s'opère à l'aide de la méthode d'itération de la puissance (voir l'Algorithme 1 pour le détail de cette méthode) afin d'en déduire le flux de neutrons ψ et la valeur propre dominante k_{eff} en itérant sur le terme de fission présenté par l'équation suivante :

$$H\psi^{g,n,(i+1)} = \frac{1}{k^{(i)}} F\psi^{g,n,(i)} \quad \text{où} \quad k^{(i+1)} = k^{(i)} \sqrt{\frac{\langle F\psi^{g,n,(i+1)}, F\psi^{g,n,(i+1)} \rangle}{\langle F\psi^{g,n,(i+1)}, F\psi^{g,n,(i)} \rangle}}$$

avec g le groupe, n pour dire que le flux est évalué dans la direction $\vec{\Omega}_n$ et (i) la i -ème itération de la puissance. Les itérations multigroupes consistent à calculer *la diffusion des neutrons des différents groupes dans le groupe considéré* [17] par le terme de la source de diffusion. Pour plus de lisibilité, en omettant l'indice i , ce problème peut s'écrire de la manière suivante :

$$\begin{pmatrix} H^{11} & H^{12} & \dots & H^{1G} \\ H^{21} & H^{22} & \dots & H^{2G} \\ \vdots & \vdots & \ddots & \vdots \\ H^{G1} & H^{G2} & \dots & H^{GG} \end{pmatrix} \begin{pmatrix} \psi^{1,n} \\ \psi^{2,n} \\ \vdots \\ \psi^{G,n} \end{pmatrix} = \begin{pmatrix} F^1 \\ F^2 \\ \vdots \\ F^G \end{pmatrix},$$

où

$$\psi^{g,n}(\vec{r}) = \int_{E_g}^{E_{g-1}} \psi(\vec{r}, E, \vec{\Omega}_n) dE$$

et

$$H^{gg}\psi^{g,n}(\vec{r}) = \vec{\Omega}_n \cdot \vec{\nabla} \psi^{g,n}(\vec{r}) + \Sigma_t^g(\vec{r}) \psi^{g,n}(\vec{r}) - \frac{1}{2\pi} \sum_{l=0}^L \frac{2l+1}{2} \sum_{m=-l}^{+l} \mathcal{R}_l^m(\vec{\Omega}_n) \Sigma_{sl}^{g \rightarrow g}(\vec{r}) \phi_{lm}^g(\vec{r})$$

$$H^{g'g}\psi^{g',n}(\vec{r}) = -\frac{1}{2\pi} \sum_{l=0}^L \frac{2l+1}{2} \sum_{m=-l}^{+l} \mathcal{R}_l^m(\vec{\Omega}_n) \Sigma_{sl}^{g' \rightarrow g}(\vec{r}) \phi_{lm}^{g'}(\vec{r}), \quad \text{si } g' \neq g$$

$$F^{g' \rightarrow g}(\vec{r}) = \frac{\chi^g}{4\pi} \nu \Sigma_f^{g'}(\vec{r}) \phi_{00}^{g'}(\vec{r})$$

Cela permet d'écrire le système suivant :

$$H^{gg} \psi^{g,n} + \sum_{g' \neq g} H^{g'g} \psi^{g',n} = \frac{1}{k_{eff}} \sum_{g'=1}^G F^{g' \rightarrow g} \psi^{g'} \quad (1.40)$$

Notons $S_g = \sum_{g'=1}^G F^{g' \rightarrow g} \psi^{g'}$, le calcul de la source monogroupe s'écrit alors :

$$H^{gg} \psi^{g,n} = \frac{1}{k_{eff}} S_g - \sum_{g' \neq g} H^{g'g} \psi^{g',n} \quad (1.41)$$

Ce système linéaire multigroupe est résolu par le biais d'une méthode de Gauss-Seidel (voir l'Algorithme 2). Enfin, les itérations internes traitent de la dépendance de la source de diffusion d'un groupe dans lui-même et correspondent à une méthode de Richardson. Au cours d'une itération interne, le flux angulaire est calculé, direction par direction, élément par élément, à partir de la discrétisation spatiale de l'équation de transport, voir l'Equation (1.39). Le schéma de résolution de l'équation de transport est synthétisé dans l'Algorithme 3. Également, le balayage cellule par cellule pour une direction $\vec{\Omega}_n$ donnée est illustrée sur la Figure 1.3.

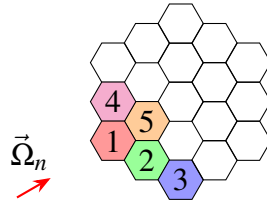


FIGURE 1.3 Illustration du balayage cellule par cellule pour une direction $\vec{\Omega}_n$ donnée.

Algorithme 1 : Algorithme de la puissance inverse

Données : $\psi^0, \nu_{\Sigma_f}, \Sigma_s, \Sigma_t$

Résultat : k_{eff}, ψ

$A = F \psi;$

tant que $\frac{|k - k^{old}|}{k^{old}} \geq \epsilon_k$ **or** $\frac{\|A - A^{old}\|}{A^{old}} \geq \epsilon_\psi$ **faire**

$H \psi = A;$

$A^{old} = A;$

$k = \sqrt{\frac{\langle A, A \rangle}{\langle A^{old}, A^{old} \rangle}};$

$A = \frac{1}{k} A;$

fin

Algorithme 2 : Algorithme de Gauss-Seidel**Données :** ψ, S_g **Résultat :** ψ **tant que non-convergence faire** **pour** $g \in [1, G]$ **faire**

$$Q = S_g - \sum_{g' \neq g} H^{gg'} \psi^{g',n}(\vec{r});$$

$$H^{gg} \psi^{g,n}(\vec{r}) = Q;$$

fin**fin****Algorithme 3 : Algorithme de résolution globale****Données :** $v\Sigma_f, \Sigma_s, \Sigma_t$ **Résultat :** k_{eff}, ψ

▷ Initialisation de l'itération externe

$$\phi^g = (1, 1, \dots, 1)^t;$$

▷ Initialisation de la source de fission de $S_g = \sum_{g'=1}^G F^{g' \rightarrow g} \phi^{g'}$ **tant que non-convergence faire**

▷ Résolution du problème multigroupe

tant que non-convergence faire **pour** $g \in [1, G]$ **faire**

▷ Calcul de la source

$$Q = S_g - \sum_{g' \neq g} \Sigma_s^{g' \rightarrow g} \phi^{g'};$$

▷ Balayage : itération flux-source

tant que non-convergence faire **pour** $\vec{\Omega}_n \in [\vec{\Omega}_0, \vec{\Omega}_N]$ **faire**

$$Q = Q_{ext} + \Sigma_s^{g \rightarrow g} \phi^g;$$

$$\vec{\Omega}_n \cdot \vec{\nabla} \psi^{g,n} + \Sigma_t \psi^{g,n} = Q;$$

fin **fin**

$$\phi^g = \sum_{n=1}^N \omega_n \psi^{g,n};$$

fin **fin** ▷ Mise à jour de la source de fission et calcul de k_{eff} **fin**

2 Exemples de bases d'éléments finis pour des polygones

Pour développer des bases d'éléments finis sur des polygones, l'un des moyens est d'utiliser la classe des fonctions coordonnées barycentriques généralisées. Historiquement, pour le triangle ou pour le parallélogramme, les coordonnées barycentriques ont servi de bases d'éléments finis d'ordre 1. Pour un polygone quelconque avec m côtés, $m \geq 4$, ces fonctions s'appellent coordonnées barycentriques généralisées et sont plus "complexes" à développer. Ainsi, au cours des dernières années, des travaux ont été réalisés pour générer des fonctions coordonnées barycentriques généralisées [38] dans l'objectif de développer des bases d'éléments finis. La section suivante en donne quelques exemples.

2.1 Quelques exemples de coordonnées barycentriques généralisées

Soit $\kappa \in \mathbb{R}^2$, un polygone avec m côtés que l'on peut noter m -gone. On note $(a_i)_{(1 \leq i \leq m)}$, ($m \geq 3$) ses sommets. Les fonctions $(v_i)_{1 \leq i \leq m}$ sont appelées coordonnées barycentriques généralisées si elles vérifient les propriétés suivantes :

$$\forall i \in \llbracket 1; m \rrbracket, \sum_{i=1}^m v_i(\vec{r}) = 1, \sum_{i=1}^m v_i(\vec{r}) a_i = \vec{r} \quad (1.42)$$

Ainsi, les fonctions coordonnées barycentriques généralisées sont des fonctions d'ordre 1. Celles-ci permettent d'effectuer une interpolation pour approcher la valeur d'une fonction f en tout point de κ . En effet, si \vec{r} est un point à l'intérieur de κ , la valeur d'une fonction f peut être évaluée de la manière suivante : $f(\vec{r}) = \sum_{i=1}^n v_i(\vec{r}) f(a_i)$

Quelques remarques importantes peuvent être ajoutées :

- Wachspress dans [103] a prouvé qu'il est impossible de trouver des fonctions coordonnées barycentriques généralisées polynomiales pour un polygone convexe avec $m \geq 4$ côtés (à part le cas du parallélogramme). Par conséquent, il n'existe pas de base nodale polynomiale pour les polygones et une preuve est donnée dans Appendix 1.
- Ces fonctions coordonnées barycentriques généralisées ne sont plus uniques pour $m \geq 4$.

De par la non unicité des fonctions coordonnées barycentriques généralisées pour des polygones avec $m \geq 4$ côtés, quelques exmples sont donnés ici.

Les fonctions "Mean value"

Ces fonctions ont été introduites par Floater dans [37]. La preuve que ces fonctions forment des fonctions coordonnées barycentriques généralisées est fournie dans [38]. Les fonctions "Mean value" $(v_i)_{1 \leq i \leq m}$ sont définies de la manière suivante :

$$\begin{cases} v_i(\vec{r}) = \frac{w_i(\vec{r})}{\sum_{j=1}^m w_j(\vec{r})} \\ \text{Où : } w_i(\vec{r}) = \frac{\tan(\alpha_{i-1}/2) + \tan(\alpha_i/2)}{\|a_i - \vec{r}\|} \end{cases} \quad (1.43)$$

Avec $\alpha_i = \alpha_i(\vec{r})$ les angles, où $0 < \alpha_i < \pi$, voir Fig. 1.4.

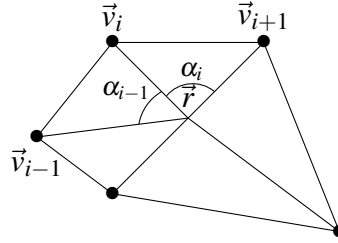


FIGURE 1.4 Notations pour les fonctions Mean value

Ces fonctions sont particulièrement adaptées aux polygones en forme d'étoile et aux polygones quelconques. En revanche, elles ne sont pas positives sur des polygones non étoilés.

Les fonctions "Maximum entropy coordinates"

Ces fonctions ont été développées par Hormann et Sukumar dans [56, 98]. Pour un point donné \vec{r} dans un m -gone convexe κ , Sukumar a proposé de trouver les fonctions coordonnées barycentriques généralisées $(v_i)_{1 \leq i \leq m}$ en maximisant l'"entropie" :

$$\begin{cases} H(\phi_1, \dots, \phi_m) = -\sum_{i=1}^m \phi_i \log(\phi_i) \\ \text{Sous les contraintes : } \sum_{i=1}^m v_i(\vec{r}) = 1, \sum_{i=1}^m v_i(\vec{r}) a_i = \vec{r} \end{cases} \quad (1.44)$$

En utilisant les multiplicateurs de Lagrange, ils ont prouvé que la solution est :

$$\begin{cases} v_i = w_i / \sum_{j=1}^m w_j \\ \text{Avec } w_i = m_i e^{\vec{\lambda} \cdot \vec{d}_i} \end{cases}$$

où $\forall i \in \llbracket 1, m \rrbracket$, (m_i) des poids positifs associés au sommet i , $\vec{d}_i = a_i - \vec{r}$ et $\vec{\lambda} \in \mathbb{R}^2$ tels que $\sum_{i=1}^m w_i(\vec{r})(\vec{d}_i) = 0$ est vérifié.

Ceci est équivalent à :

$$\text{Trouver } \vec{\lambda} \in \mathbb{R}^2, \text{ tel que, } \sum_{i=1}^m m_i e^{\vec{\lambda} \cdot \vec{d}_i} \vec{d}_i = 0 \quad (1.45)$$

L'Equation (1.45) a une unique solution car $F(\vec{\lambda}) = \sum_{i=1}^m m_i e^{\vec{\lambda} \cdot \vec{d}_i}$ est une fonction strictement convexe. Par conséquent, elle possède un unique minimum $\vec{\lambda}$; voir [38, 98] pour plus d'informations (la démonstration est basée sur la matrice hessienne). Par ailleurs, les coordonnées barycentriques $(v_i)_{1 \leq i \leq m}$ obtenues vérifient les propriétés de Lagrange.

Les fonctions de Wachspress

Ces fonctions ont été introduites pour la première fois par Wachspress dans [103]. Ici, l'objectif est d'introduire ces fonctions à l'ordre 1. Plus de détails seront fournis dans les sections suivantes. Ces fonctions s'écrivent ainsi :

$$v_i = c_i \frac{p^i}{q} \quad (1.46)$$

- p_i est un polynôme de degré $m - 2$, $p_i = \prod_{j \neq i, i+1} l_j$ avec l_j tout polynôme linéaire non trivial qui s'annule sur la droite passant par l'arête $\vec{e}_j = [a_{j-1}, a_j]$, par exemple, l_j peut-être l'équation de la droite passant par les points a_{j-1} et a_j .
- q est l'adjoint polynomial du polygone *i.e.* l'unique polynôme de degré $\leq m - 3$ qui s'annule sur les points d'intersection externe du polygone κ . Ce polynôme correspond à la forme passant par tous les points d'intersection des droites prolongeant les arêtes du polygone κ (qui peuvent être des points à l'infini pour certains polygones). $q \neq 0$ sur κ et pour les cas particuliers du triangle et du rectangle $q = 1$.
- c_i est le facteur de normalisation, $c_i = \frac{q(a_i)}{p^i(a_i)}$.

Si nous considérons Figure 1.5, alors les fonctions $(v_i)_{1 \leq i \leq m}$ sont :

$$\phi_1 = c_1 \frac{l_3 l_4}{q}, \phi_2 = c_2 \frac{l_4 l_1}{q}, \phi_3 = c_3 \frac{l_1 l_2}{q}, \phi_4 = c_4 \frac{l_2 l_3}{q}$$

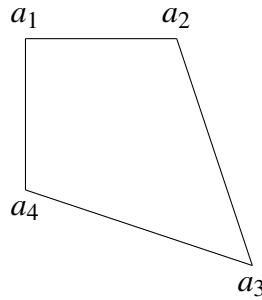


FIGURE 1.5 Un quadrilatère quelconque

D'autres exemples de fonctions pourraient être proposés, comme celles décrites par Sibson ([95], les fonctions "harmonic coordinates" dans ([106]), etc. Des informations sur les fonctions coordonnées barycentriques généralisées et leurs comparaisons peuvent être trouvées dans [3]. Le Tableau 1.1 résume quelques propriétés vérifiant certaines fonctions.

TABLEAU 1.1 Propriétés de certaines fonctions coordonnées barycentriques généralisées. Tableau tiré en partie de [3].

Fonctions	Domaine de validité	Positivité	Propriété de Lagrange	Régularité
Wachspress	polygone convexe	✓	✓	C^∞
Mean value	polygone arbitraire	x	✓	C^∞
Maximum entropy	polygone arbitraire	✓	✓	C^k
Harmonic	polygone arbitraire	✓	✓	C^∞

Bien que les fonctions Maximum entropy et Harmonic présentent l'avantage de traiter des polygones arbitraires, celles-ci ne sont pas analytiques. Or, dans notre contexte de résolution de l'équation de transport par éléments finis, il est pratique d'obtenir une solution approchée analytique que ce soit pour le calcul des matrices ou pour le post traitement. Cette démarche exclue l'usage de ces fonctions. Également, l'inconnue de l'équation de transport est le flux neutronique. Celui-ci a pour propriétés d'être C^0 sur tout le domaine (hormis dans certains cas spécifiques) et également d'être positif. Aussi, le domaine est maillé avec des hexagones qui est un polygone convexe. Par conséquent, les fonctions de Wachspress semblent adaptées à notre problématique.

Un objectif clé de ce travail consiste à construire des bases d'éléments finis d'ordre élevé. Or, les fonctions coordonnées barycentriques généralisées sont des fonctions d'ordre 1 et la génération d'une base de fonction d'ordre p quelconque à partir de ces dernières n'est pas une trivialité. En particulier, Rand et Gillette dans [83] ont développé une méthode générique pour construire des éléments de Sérendip d'ordre 2 à partir de fonctions coordonnées barycentriques généralisées. Ces éléments de Sérendip ont ensuite été utilisés par Hackemack et Ragusa [51] pour résoudre l'équation de transports sur des polygones. Toutefois, il n'existe pas de méthode générale pour produire des bases de fonctions d'ordre supérieur à 2 sur des polygones. Ainsi, pour monter en ordre, un point de départ a été de reprendre les travaux de Gout dans sa thèse [46] où il y décrit des bases d'éléments finis de Wachspress sur différents polygones dont l'hexagone aux ordres 1 à 3. Cela étant réalisé dans l'idée de pouvoir ensuite généraliser ces travaux. Ainsi dans le chapitre 2, nous nous focaliserons sur le développement de ces fonctions de base à des ordres supérieurs ou égal à 3.

3 Synthèse

Dans ce chapitre, nous avons introduit l'équation de transport et les discrétisations considérées pour les variables énergétique, angulaire et spatiale respectivement au travers des méthodes multigroupes, S_n et de la méthode de Galerkin discontinue.

La discrétisation spatiale est le cœur de ces travaux, ainsi différentes bases d'éléments finis ont été présentées pour traiter un maillage hexagonal. En premier lieu, celles issues des fonctions coordonnées barycentriques généralisées. Dans ce cadre, une première introduction aux fonctions de Wachspress a été réalisée.

Par la suite, l'objectif est alors de reprendre les travaux de thèse de Gout [46] où plusieurs bases d'éléments finis de Wachspress sont décrites sur différents polygones dont l'hexagone aux ordres 1 à 3. À partir de ces résultats, l'idée est ensuite de développer une méthode afin d'obtenir des fonctions de Wachspress à un ordre k ¹ pour l'hexagone régulier.

Aussi, la discrétisation DG permet d'utiliser des bases de fonction polynomiales comme les polynômes de Legendre (l'Equation (1.21)) ou des bases de monômes sur l'hexagone. Ces bases modales sont déjà utilisées pour la résolution de l'équation de transport dans le solveur APOLLO3[®]/NYMO. Elles constituent alors une alternative aux GBC. Par la suite, nous développerons et utiliserons des bases de monômes orthogonalisées.

Ainsi, dans l'objectif de développer un solveur de transport sur des maillages hexagonaux, nous allons développer deux types bases d'éléments finis : des base de monômes orthogonalisées et des bases de Wachspress. Ces deux bases seront ensuite comparées via les erreurs d'approximation entre une solution analytique et la solution approchée par la méthode des solutions manufacturées. D'autres comparaisons seront réalisées sur des cas réacteurs en analysant les résultats sur le k_{eff} et sur les taux d'absorption.

1. L'ordre k doit être inférieur au nombre de côtés du polygone, ce point sera détaillé par la suite.

Chapitre 2

Développement de bases d'éléments finis sur des polygones convexes

Dans l'optique de résoudre l'équation de transport sur des maillages hexagonaux, des bases d'éléments finis ont été développées, plus particulièrement, des bases nodales de fonction de Wachspress et des bases modales de monômes orthogonalisées. L'objectif étant de les comparer numériquement au travers d'erreurs d'approximation.

Pour la base de fonctions de Wachspress, celles-ci permettent de construire des éléments finis de Lagrange pour un polygone convexe κ avec m côtés. Elles sont en particulier C^∞ sur κ et définissent une base de fonction C^0 sur tout le domaine \mathcal{D} . De plus, pour des bases de Wachspress d'ordre k , $k < m$, les degrés de libertés (*i.e.* les nœuds) se situent essentiellement sur le bord de κ . Ainsi, les fonctions de base peuvent être assimilées à des fonctions de Sérendip [51]. Le développement de ces fonctions de Wachspress est le point central de ce chapitre car cette partie a nécessité la conception d'une nouvelle méthode. Ce travail a par la suite été valorisée dans une publication pour le "Journal of Computational Physics" [62].

À partir des résultats de Wachspress [103], Gout décrit des bases d'éléments finis rationnelles pour différents polygones. Gout a notamment proposé différentes bases de Wachspress pour les quadrangles, pentagones et hexagones pour certains ordres, en particulier jusqu'à l'ordre 3 pour l'hexagone. Bien que des preuves rigoureuses aient été fournies *a posteriori* pour vérifier que ces fonctions forment des bases d'éléments finis, aucune méthode n'est fournie pour construire de telles bases à un ordre supérieur ou sur un polygone différent.

Ainsi, ce chapitre étend les travaux de Gout et de Wachspress en proposant une méthode originale pour calculer des bases de Wachspress sur tout type de polygones convexes (avec m

côtés) pour tout ordre $k < m$ à partir des propriétés qu'elles doivent vérifier (*i.e.* s'assurer que ces bases de fonctions forment un élément fini de Lagrange). Cette méthode de génération de bases a ensuite été appliquée à deux polygones convexes : l'hexagone régulier et un pentagone irrégulier. Pour l'hexagone régulier, des bases de Wachspress ont été obtenues jusqu'à l'ordre 5 et jusqu'à l'ordre 4 pour le pentagone irrégulier.

Concernant les bases modales de monômes orthogonalisées, ces dernières sont couramment utilisées dans le cadre DG [57, 87]. Cependant, elles ne peuvent être appliquées pour la méthode des éléments finis continus. En effet, la continuité aux interfaces entre les mailles n'est pas vérifiée. Ici, nous présenterons l'implémentation et le processus d'orthogonalisation de ces bases pour un hexagone.

1 Forme générale des fonctions rationnelles de Wachspress sur des polygones convexes à un ordre $k < m$

Pour former des éléments finis C^0 sur un domaine \mathcal{D} , Wachspress dans [103] décrit les propriétés que doivent vérifier les fonctions de base. Elles peuvent être brièvement résumées comme suit :

- Chaque fonction de base est associée à un nœud du polygone.
- Ces fonctions sont normalisées pour ce nœud, *i.e.* égales à 1.
- Chaque fonction s'annule sur les côtés qui n'intersectent pas ce nœud.

Lorsque l'ordre augmente, les fonctions de bases vont nécessiter des nœuds supplémentaires (autres que les sommets du polygone). Dans le contexte des éléments de Sérendip, ces nœuds sont placés sur le bord du polygone. C'est-à-dire que pour une fonction donnée, celle-ci doit s'annuler sur tous les nœuds hormis celui sur lequel elle est "ancrée", nous reviendrons sur cette notion par la suite. Il est important de noter qu'en ne considérant exclusivement les nœuds sur les bords du polygone, cette approche se limite à un ordre k tel que $k < m$. En effet, comme l'a démontré Wachspress [103], pour les ordres supérieurs, des points intérieurs sont nécessaires pour définir les bases associées. Un exemple de preuve est fournie Annexe 2. Dans ces travaux, nous ne considérons que le cas $k < m$. Cela se justifie par notre application, c'est-à-dire la résolution de l'équation de transport sur des maillages hexagonaux réguliers. Dans ce cadre, afin d'améliorer le rapport temps de calcul et précision, il est probablement peu intéressant de monter au-delà de l'ordre 5, et si tel était le cas, il faudrait plutôt recourir à un raffinement de l'hexagone en triangles ou losanges. Précisons que pour l'ordre 5, l'espace d'approximation contient trente fonctions de base.

1.1 Des premiers travaux réalisés par Gout

Ici, nous nous intéressons seulement aux bases proposées par Gout pour l'hexagone régulier. Pour ce cas particulier, les éléments géométriques suivants sont donnés comme indiqué dans la Figure 2.1 et suivant les mêmes notations que dans [46, 47].

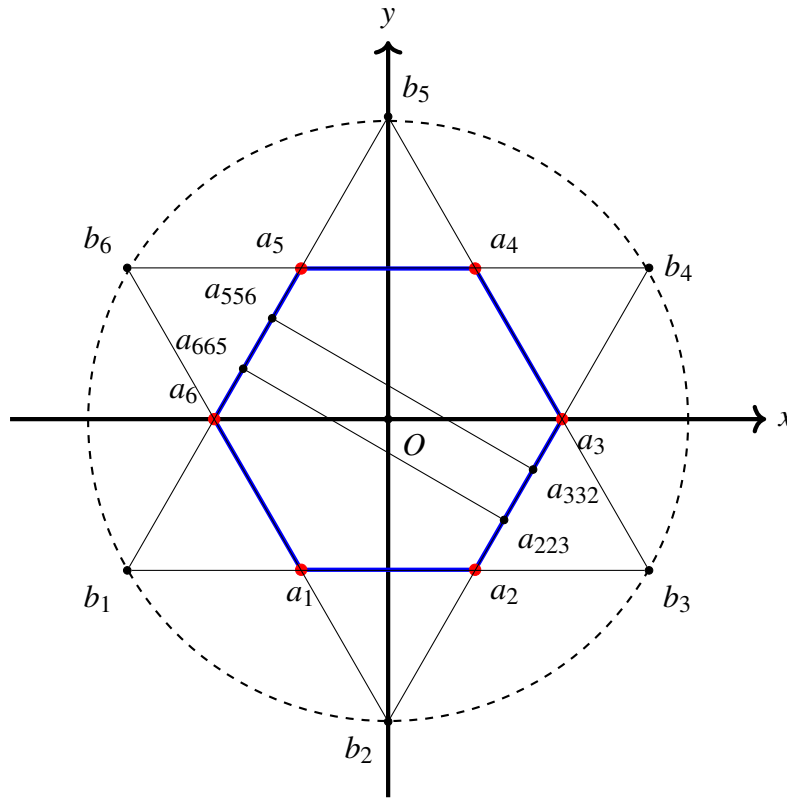


FIGURE 2.1 Notations introduites par Gout dans [46] pour décrire les éléments géométriques nécessaires pour définir des fonctions de Wachspress d'ordre 3

Considérons un hexagone κ orienté avec une face vers le bas¹ de centre $(0,0)$, de pas (distance entre deux côtés parallèles) p . La longueur d'une arête est donc de $\frac{p}{\sqrt{3}}$. Les sommets a_i de l'hexagone sont numérotés comme indiqué sur la Figure 2.1, de sorte que a_i et a_{i+1} soient consécutifs. Soit $(x,y) \in \kappa$, pour tout i dans $\mathbb{Z}/6\mathbb{Z}$, $l_i(x,y) = 0$ est l'équation de la droite d_i passant par a_{i-1} et a_i . L'intersection des droites d_{i-1} et d_{i+1} est le point b_i , aussi appelé point d'intersection externe (EIP : External Intersection Point), et $q(x,y) = 0$ est l'équation du cercle Q passant par tous les points b_i (on l'appelle aussi le cercle adjoint de l'hexagone régulier).

1. Par opposition avec une orientation avec un sommet pointu vers le bas.

Ces éléments géométriques sont suffisants pour définir les fonctions de base de Wachspress d'ordre 1 que l'on note w_i , ces fonctions étant ancrées aux sommets de l'hexagone :

$$w_i = c_i \frac{l_{i+2}l_{i+3}l_{i+4}l_{i+5}}{q} \quad (2.1)$$

où c_i est une constante de normalisation, *i.e.* $w_i(a_i) = 1$.

Pour définir des fonctions de Wachspress d'ordre 2, de nouveaux nœuds sont nécessaires. C'est ainsi que l'on introduit $a_{i,i+1}$, le point médian entre a_i et a_{i+1} . On note également $l'_i(\vec{x}) = 0$ l'équation de la droite d'_i passant par $a_{i-1,i}$ et $a_{i,i+1}$. En utilisant ces constructions géométriques, et celles définies précédemment, il est maintenant possible de construire les fonctions de base de Wachspress d'ordre deux :

$$\begin{cases} w_i &= c_i \frac{l_{i+2}l_{i+3}l_{i+4}l_{i+5}l'_i}{q} \\ w_{i,i+1} &= c_{i,i+1} \frac{l_{i+2}l_{i+3}l_{i+4}l_{i+5}l_i}{q} \end{cases} \quad (2.2)$$

où c_i et $c_{i,i+1}$ sont choisis de telle sorte que $w_i(a_i) = 1$ et $w_{i,i+1}(a_{i,i+1}) = 1$.

Enfin, pour définir l'ordre 3, Gout introduit les nœuds suivants $a_{i+1,i+1,i}$ (respectivement $a_{i,i,i+1}$) se calculant comme $\frac{1}{3}(2a_{i+1} + a_i)$ (respectivement $\frac{1}{3}(2a_i + a_{i+1})$). L'équation des droites passant par $a_{i+1,i+1,i}$ et $a_{i+3,i+3,i+4}$ se note $l'_{i+1,i+3}(\vec{x}) = 0$. Enfin, $\gamma(\vec{x}) = 0$ est l'équation du cercle Γ passant par les 12 points $a_{i+1,i+1,i}$ et $a_{i,i,i+1}$. Avec ce dernier ensemble d'éléments géométriques, les fonctions de base de Wachspress d'ordre 3 sont définies :

$$\begin{cases} w_i &= c_i \frac{l_{i+2}l_{i+3}l_{i+4}l_{i+5}\gamma}{q} \\ w_{i,i,i+1} &= c_{i,i,i+1} \frac{l_{i+2}l_{i+3}l_{i+4}l_{i+5}l'_{i+1,i+3}}{q} \\ w_{i+1,i+1,i} &= c_{i+1,i+1,i} \frac{l_{i+2}l_{i+3}l_{i+4}l_{i+5}l'_{i,i+1}}{q} \end{cases} \quad (2.3)$$

où c_i , $c_{i,i,i+1}$ et $c_{i+1,i+1,i}$ sont choisis de telle sorte que $w_i(a_i) = 1$, $w_{i,i,i+1}(a_{i,i,i+1}) = 1$ et $w_{i+1,i+1,i}(a_{i+1,i+1,i}) = 1$ respectivement.

La Figure 2.2 montre les fonctions de base à l'ordre 3 pour $i = 2$. Ces fonctions sont clairement $C^\infty(\kappa)$ et vérifient la propriété de Lagrange.

Gout a prouvé que ces fonctions rationnelles forment une base, notons P l'espace vectoriel engendré par ces fonctions tel qu'à l'ordre k , $P_k(\kappa) \subset P$ et $P_{k+1}(\kappa) \not\subset P$ [103, 47] où P_k est l'espace des polynômes de degré inférieur ou égal à k . En notant Σ l'ensemble des nœuds, le triplet (κ, P, Σ) forme alors un élément fini C^0 d'ordre k au sens de Ciarlet *cf.* [21].

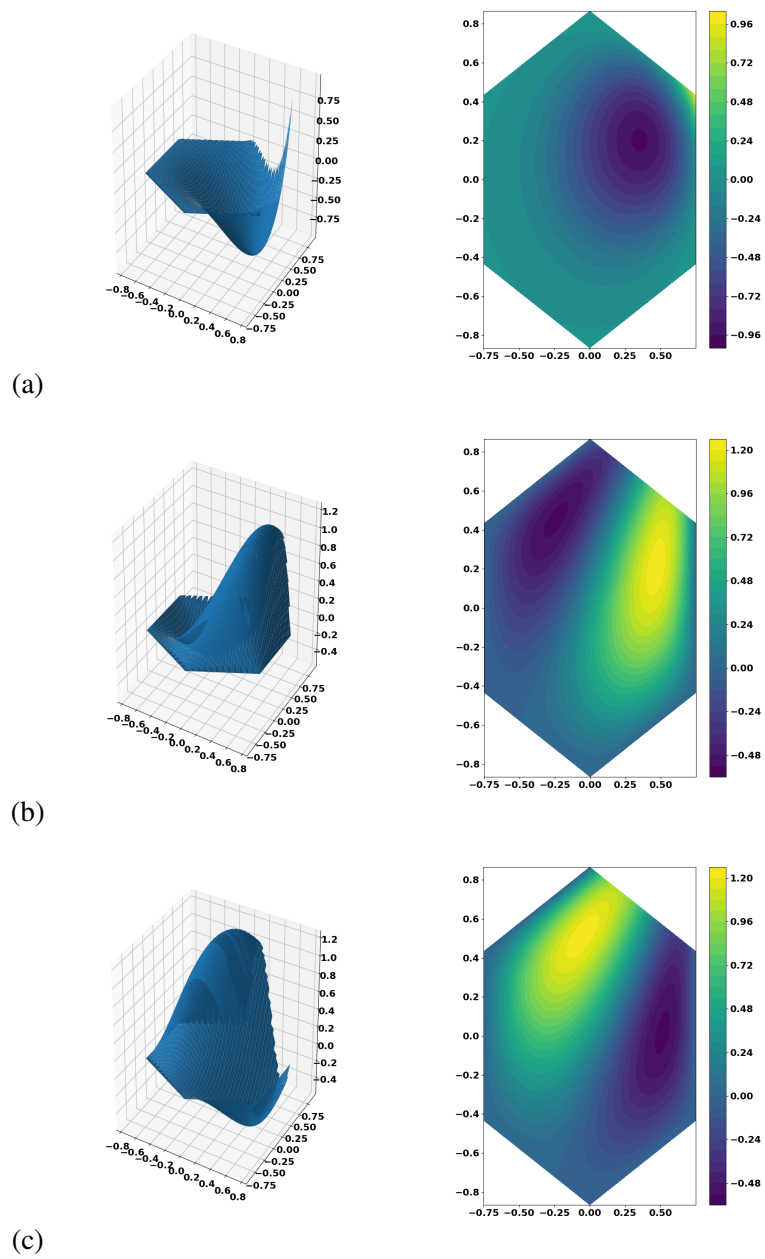


FIGURE 2.2 Exemple de fonctions de Wachspress à l'ordre 3.

Ces bases de fonctions proposées par Gout ont servi de point d'entrée pour la résolution de l'équation de transport sur des maillages hexagonaux [16] (jusqu'à l'ordre 3). Également, de par l'infinité de bases de Wachspress possibles pour des ordres supérieurs ou égaux à 3², il sera intéressant de les comparer au travers de résultats numériques.

Ce chapitre étend ainsi les travaux de Gout et de Wachspress en proposant une méthode originale pour calculer des bases de Wachspress sur tout type de polygones convexes (avec m côtés) pour tout ordre $k < m$ à partir des propriétés qu'elles doivent vérifier (*i.e.* s'assurer que ces bases de fonctions forment un élément fini de Lagrange).

1.2 Extension des travaux de Gout à des ordres élevés

Notations

Pour présenter notre méthode, nous changeons les notations introduites par Gout pour faciliter l'écriture des fonctions de Wachspress. Nous désignons par κ un polygone strictement convexe de m côtés avec les sommets : $\{a_i\}_{i \in \llbracket 1; m \rrbracket}$, tels que a_i et a_{i+1} soient consécutifs.³ Les coordonnées de a_i dans le plan $x - y$ sont notées (x_i, y_i) .

Pour un ordre k donné, nous désignons également les nœuds équidistants sur l'arête $[a_i, a_{i+1}]$ par $\{a_{i_j}\}_{j \in \llbracket 1; k-1 \rrbracket}$ avec $a_{i_j} = \frac{1}{k}((k-j)a_i + ja_{i+1})$. Les coordonnées de a_{i_j} dans le plan $x - y$ sont notées (x_{i_j}, y_{i_j}) . Ces nœuds constituent les "points d'ancrage" des fonctions de bases pour le côté compris entre a_i et a_{i+1} .

Aussi, l'équation de la droite d_i passant par a_{i-1} et a_i est donnée par $l_i = 0$ et $B = \{b_i\}_{i \in \llbracket 1; m \rrbracket}$, est l'ensemble des différents points d'intersection des droites $\{d_i\}_{i \in \llbracket 1; m \rrbracket}$.

L'exemple de l'hexagone régulier ($m = 6$) à l'ordre $k = 3$ est représenté sur la Figure 2.3.

2. Ce point sera détaillé par la suite.

3. Pour des raisons de concision, les notations associées à l'arithmétique modulaire ne sont pas utilisées et on considère implicitement que, $\forall i \in \llbracket 1; m \rrbracket, \forall n \in \mathbb{Z} a_{i+nm}$ désigne a_i .

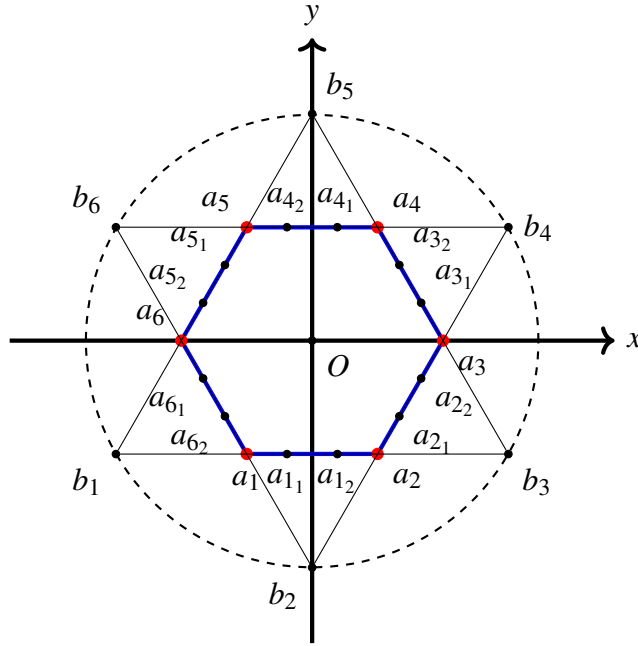


FIGURE 2.3 Eléments géométriques pour définir les fonctions de Wachspress sur un hexagone régulier à l'ordre 3.

Définition des fonctions rationnelles

Pour un ordre k donné tel que $k < m$, selon Wachspress [103], $k \times m$ fonctions rationnelles w doivent être définies pour obtenir une base sur κ . Suivant [103], $\forall i \in \llbracket 1; m \rrbracket$, $\forall j \in \llbracket 1; k-1 \rrbracket$, les fonctions de base peuvent s'écrire de manière plus générale comme :

$$w_i = c_i \frac{p^i r^i}{q} \quad (2.4)$$

$$w_{i_j} = c_{i_j} \frac{p^{i_j} r^{i_j}}{q} \quad (2.5)$$

où

- p^i (resp. p^{i_j}) est le produit des arêtes de κ ne passant pas par a_i (resp. a_{i_j}).
- r^i (resp. r^{i_j}) est une fonction polynomiale d'ordre $k-1$ (resp. $k-2$) qui vérifie $\forall l \in \llbracket i-1, i \rrbracket, r^i(a_{l_j}) = 0$ (resp. $\forall h \neq j, r^{i_j}(a_{i_h}) = 0$).
- q est le polynôme adjoint du polygone tel que $\forall b \in B, q(b) = 0$. Pour un polygone régulier, il correspond à une équation de cercle. Dans un cas général, q est de degré maximal $m-3$.
- c_i (resp. c_{i_j}) est le coefficient de normalisation tel que $w_i(a_i) = 1$ (resp. $w_{i_j}(a_{i_j}) = 1$).

De plus, on note :

- $P = \text{Vect}(\{\forall i \in \llbracket 1; m \rrbracket, \forall j \in \llbracket 1; k-1 \rrbracket, w_i = c_i \frac{p^i r^i}{q}; w_{ij} = c_{ij} \frac{p^i r^j}{q}\})$, l'espace des fonctions de base ou espace d'approximation.
- $\Sigma = \{a_i; a_{ij}; \forall i \in \llbracket 1; m \rrbracket, \forall j \in \llbracket 1; k-1 \rrbracket\}$, l'ensemble des degrés de liberté.

Le triplet (κ, P, Σ) forme alors un élément fini d'ordre k au sens de Ciarlet cf. [21]. Ainsi, il vérifie les propriétés suivantes :

$$\begin{cases} (\kappa, P, \Sigma) \text{ est } C^0 \\ P_k(\kappa) \subset P, P_{k+1}(\kappa) \not\subset P \end{cases} \quad (2.6)$$

Si les fonctions p^i et p^{ij} peuvent facilement être déduites (car correspondant à des produits de droites), la difficulté est tout autre pour les fonctions r^i et r^{ij} . En effet, ces fonctions sont loin d'être triviales à obtenir car les seules informations à disposition pour les calculer sont : 1) la relation d'inclusion suivante : $P_k \subset P, P_{k+1} \not\subset P$, 2) les propriétés de Lagrange que doivent vérifier les fonctions de Wachspress. Les seuls cas "simples" correspondent aux ordres 1 et 2. Au-delà de ces ordres, la littérature ne fournit que des exemples pour des cas particuliers, notamment par Gout pour les cas spécifiques de l'hexagone régulier et du quadrilatère [46].

Il est important de noter que dans le cas d'un polygone irrégulier, le calcul de l'adjoint q n'est pas une trivialité. Ainsi, des travaux ont été proposés par Dasgupta dans [28] ou Powar dans [81] pour remédier à ce problème. Dans [28], Dasgupa présente l'algorithme GADJ qui se résume en la construction d'un système linéaire dont les inconnues sont les coefficients de la fonction q . En effet, q est un polynôme de degré connu en fonction du polygone. Le système linéaire obtenu est basé sur les propriétés géométriques que doivent vérifier les fonctions de Wachspress et du degré de q . Pour construire ce système linéaire, l'idée est d'utiliser le numérateur des fonctions de Wachspress à l'ordre 1 car pour cet ordre, les fonctions de base sont connues. Ainsi, en résolvant ce système linéaire, l'adjoint q est obtenu.

Dans ces travaux de recherche, la démarche est sensiblement la même pour générer les fonctions r^i, r^{ij} . En effet, à partir des propriétés que doivent vérifier les fonctions de Wachspress et de la connaissance du dénominateur, un système linéaire dont les inconnues sont les coefficients des fonctions r^i, r^{ij} est construit et résolu. De manière plus générale, en combinant notre méthode de génération de base et celle de Dasgupta, il est alors possible de développer des fonctions de Wachspress pour n'importe quel polygone convexe avec m

côtés à un ordre k , $k < m$. La section suivante détaille la procédure calculant les coefficients associés aux fonctions r^i et r^{ij} .

2 Une méthode de génération de bases de fonctions rationnelles

Nous considérons ici que l'adjoint q est connu (par la méthode de Dasgupta par exemple [28]), nous allons alors calculer le numérateur des fonctions de Wachspress, plus particulièrement les fonctions polynomiales r^i et r^{ij} à des ordres k , tels que $1 < k < m$.

2.1 Principe général

Cette approche consiste à utiliser le calcul symbolique pour implémenter les équations que doivent vérifier les fonctions de Wachspress. Plus particulièrement, les contraintes suivantes doivent être vérifiées :

— Contraintes géométriques (avec $\delta_{i,j}$, le symbole de Kronecker) :

$$\left\{ \begin{array}{l} \forall i, l \in \llbracket 1; m \rrbracket, w_i(a_l) = \delta_{i,l} \\ \forall l \in \{i, i-1\}, \forall j \in \llbracket 1; k-1 \rrbracket, w_i(a_{l_j}) = 0 \\ \forall i, l \in \llbracket 1; m \rrbracket, \forall j \in \llbracket 1; k-1 \rrbracket, w_{i_j}(a_l) = 0 \\ \forall i \in \llbracket 1; m \rrbracket, \forall j, p \in \llbracket 1; k-1 \rrbracket, w_{i_j}(a_{i_p}) = \delta_{j,p} \end{array} \right. \quad (2.7)$$

— Projection exacte de chaque monôme de P_k :

$$x^\alpha y^\beta = \sum_{i=1}^n (w_i(x, y) x_i^\alpha y_i^\beta) + \sum_{j=1}^{k-1} w_{i_j}(x, y) x_{i_j}^\alpha y_{i_j}^\beta \quad (2.8)$$

avec $\forall (\alpha, \beta) \in C_k = \{(\alpha, \beta) \in \llbracket 0, k \rrbracket^2, \alpha + \beta \leq k\}$

Grâce au calcul symbolique, ces équations conduisent à un système linéaire dont les inconnues sont les coefficients de r^i et r^{ij} (contenus dans le vecteur \vec{x}) sous la forme :

$$A\vec{x} = \vec{b} \quad (2.9)$$

avec A une matrice rectangulaire et \vec{b} un vecteur. Ces derniers sont obtenus à l'aide d'une identification. Détaillons ce processus d'identification.

Premièrement, pour un ordre $k < m$, introduisons les cf_i^l et $cf_{i_j}^l$ respectivement les coefficients de r^i et r^{i_j} , tels que :

$$\begin{aligned} r^i(x, y) &= \sum_{l=1}^{\frac{k(k+1)}{2}} cf_i^l x^{\alpha_l} y^{\beta_l} \\ r^{i_j}(x, y) &= \sum_{l=1}^{\frac{(k-1)k}{2}} cf_{i_j}^l x^{\alpha'_l} y^{\beta'_l} \end{aligned} \quad (2.10)$$

où $(\alpha_l, \beta_l) \in C_{k-1}$ et $(\alpha'_l, \beta'_l) \in C_{k-2}$

Réalisons une identification dans le cas d'une base 3. Notons \vec{x} notre vecteur d'inconnues, $\vec{x} = [cf_1^1, \dots, cf_6^6, cf_{1_1}^1, \dots, cf_{6_3}^3]$

Projetons dans la base de Wachspress, chaque monôme $x^\alpha y^\beta \in P_3$, avec $(\alpha, \beta) \in C_3$:

$$\left\{ \begin{array}{l} \sum_{i=1}^m (w_i(x, y) + w_{i_1}(x, y) + w_{i_2} = 1 \\ \sum_{i=1}^m (w_i(x, y)x_i + w_{i_1}(x, y)x_{i_1} + w_{i_2}(x, y)x_{i_2} = x \\ \vdots \\ \sum_{i=1}^m (w_i(x, y)y_i^3 + w_{i_1}(x, y)y_{i_1}^3 + w_{i_2}(x, y)y_{i_2}^3 = y^3 \end{array} \right. \quad (2.11)$$

Une démarche similaire peut être réalisée pour construire un ensemble d'équations à partir des contraintes géométriques.

Cependant, en pratique, une manipulation algébrique est nécessaire pour normaliser les fonctions afin d'éviter des relations non linéaires entre les différents termes des fonctions de Wachspress. En effet, par définition, les coefficients c_i sont définis comme :

$$w_i(a_i) = 1 \Leftrightarrow c_i = \frac{q(a_i)}{p^i(a_i)r^i(a_i)}$$

De sorte qu'une dépendance non linéaire apparaît entre les coefficients de r^i et c_i . Pour résoudre ce problème, nous modifions légèrement le cadre du problème en imposant les coefficients inconnus comme des coefficients normalisés *a priori* en considérant les fonctions $\tilde{r}^i = \frac{r^i}{r^i(a_i)}$ et $\tilde{r}^{i_j}(x, y) = \frac{r^{i_j}}{r^{i_j}(a_{i_j})}$. Ainsi, les fonctions requises sont effectivement normalisées

de telle sorte que les fonctions soient égales à 1 sur les nœuds d'ancrage. Par conséquent :

$$\tilde{r}^i(x, y) = \sum_{l=1}^{\frac{k(k+1)}{2}} c_{f_i^l}^l x^{\beta_l} y^{\gamma_l} \text{ avec } c_{f_i^l}^l = \frac{c_{f_i^l}}{\sum_{m=1}^{\frac{k(k+1)}{2}} c_{f_i^m}^m x_i^{\beta_m} y_i^{\gamma_m}} \quad (2.12)$$

$$\tilde{r}^{ij}(x, y) = \sum_{l=1}^{\frac{k(k+1)}{2}} c_{f_{ij}^l}^l x^{\beta_l} y^{\gamma_l} \text{ avec } c_{f_{ij}^l}^l = \frac{c_{f_{ij}^l}}{\sum_{m=1}^{\frac{k(k+1)}{2}} c_{f_{ij}^m}^m x_{ij}^{\beta_m} y_{ij}^{\gamma_m}} \quad (2.13)$$

Ainsi, les projections des Equations (2.11) s'écrivent en réalité :

$$\left\{ \begin{array}{l} \sum_{i=1}^m \left(\frac{q(a_i)}{p^i(a_i)} p^i(x, y) \tilde{r}^i(x, y) + \sum_{j=1}^2 \frac{q(a_{ij})}{p^{ij}(a_{ij})} p^{ij}(x, y) \tilde{r}^{ij}(x, y) \right) = q(x, y) \\ \sum_{i=1}^m \left(\frac{q(a_i)}{p^i(a_i)} p^i(x, y) \tilde{r}^i(x, y) x_i + \sum_{j=1}^2 \frac{q(a_{ij})}{p^{ij}(a_{ij})} p^{ij}(x, y) \tilde{r}^{ij}(x, y) x_{ij} \right) = q(x, y) x \\ \vdots \\ \sum_{i=1}^m \left(\frac{q(a_i)}{p^i(a_i)} p^i(x, y) \tilde{r}^i(x, y) y_i^3 + \sum_{j=1}^2 \frac{q(a_{ij})}{p^{ij}(a_{ij})} p^{ij}(x, y) \tilde{r}^{ij}(x, y) y_{ij}^3 \right) = q(x, y) y^3 \end{array} \right. \quad (2.14)$$

Les fonctions des membres de gauche et de droite étant des polynômes, une identification peut être réalisée pour chaque monôme. Cela permet de créer le vecteur \vec{b} . Aussi, les termes intervenant devant les coefficients de $c_{f_i^l}^l$ sont stockés dans une matrice A . Ainsi, on peut alors écrire : $A\vec{x} = \vec{b}$

À ce stade, plusieurs remarques peuvent être faites :

- Les équations des contraintes géométriques sont incluses dans les équations de projection des monômes, une démonstration a été réalisée et est fournie par la suite.
- Ce système semble sous-déterminé dans le cas général, *i.e.* la solution n'est pas unique. Ces résultats ont été observés pour l'hexagone régulier et le pentagone irrégulier à un ordre $k \geq 3$.

De par le caractère sous-déterminé du système linéaire, une analyse approfondie est effectuée afin de distinguer les variables dépendantes et indépendantes pour déduire toutes les solutions possibles. À cette fin, la matrice augmentée $[A; \vec{b}]$ du système linéaire est écrite

dans sa forme échelonnée réduite (**rref** : reduced row echelon form)⁴. L'algorithme est détaillé dans [97].

La matrice rref peut être utilisée pour distinguer les variables dépendantes et indépendantes ; formellement, une matrice de permutation Q peut être définie de sorte que $[\vec{x}_d, \vec{x}_i]^T = Q\vec{x}$ où \vec{x}_d (resp. \vec{x}_i) est un vecteur de longueur notée n_d (resp. n_i) contenant les variables dépendantes (resp. libres). Par conséquent, le système linéaire peut être reformulé sous la forme suivante :

$$A_d \vec{x}_d = \vec{b}_d - A_i \vec{x}_i \quad (2.15)$$

où A_d est une matrice inversible $n_d \times n_d$, A_i une matrice $n_d \times n_i$ et \vec{b}_d la partie du membre de droite associée aux variables dépendantes.

Le système linéaire étant sous-déterminé, il existe un nombre infini de solutions en fonction des choix relatifs aux variables indépendantes. Dans [46], ces possibles degrés de liberté n'ont pas été anticipés et des formes arbitraires de r^i et r^{ij} étaient imposées. Ainsi, Gout n'aurait pas été en mesure de calculer des bases à des ordres supérieurs à 3, car les fonctions r^i et r^{ij} sont des polynômes quartiques et cubiques. Ces variables libres pourraient être utilisées pour imposer des propriétés supplémentaires aux bases (par exemple, des symétries), si nécessaire. Une fois les variables indépendantes définies, les variables dépendantes pourraient être obtenues en résolvant l'Equation (2.15). Une fois cette solution obtenue, nous nous assurons qu'elle vérifie les contraintes (issues des Equations 2.7) et la projection des monômes de (Equation (2.8)) telle que les erreurs associées (définies dans 1 et 2) sont inférieures à une précision donnée.

Toutefois, il est important de noter que pour l'hexagone régulier et le pentagone irrégulier, la matrice A est mal conditionnée, son conditionnement est supérieur à 10^{14} . Ce mauvais conditionnement explique pourquoi, pour certaines tolérances, il est impossible d'obtenir la matrice $[A, \vec{b}]$ sous forme échelonnée réduite.

Une fois la tolérance trouvée pour calculer la matrice échelonnée réduite, la matrice A_d obtenue est aussi mal conditionnée⁵. On observe alors que les systèmes avec ou sans prise en compte des contraintes géométriques, bien qu'équivalent mathématiquement, ne le sont pas numériquement et conduisent à des solutions qui peuvent différer significativement.

4. Une matrice est sous forme échelonnée réduite lorsqu'elle satisfait les conditions suivantes : 1) Le premier élément non nul de chaque ligne, appelé entrée principale, est égal à 1. 2) Chaque entrée principale se trouve dans une colonne à droite de l'entrée principale de la ligne précédente. 3) Les rangées comportant tous les éléments nuls, le cas échéant, se trouvent sous les rangées comportant un élément non nul.

5. On notera cependant que le conditionnement de la matrice A du système linéaire est 10^3 fois plus grand que celui de A_d . Par conséquent le passage par la matrice sous forme échelonnée réduite est une piste d'amélioration pour améliorer le conditionnement.

Cependant, l'ajout des contraintes géométriques dans le système est avantageux pour améliorer la précision pour laquelle ces équations sont vérifiées. En effet, la base résultante de l'ajout des contraintes dans le système linéaire permet de gagner un à trois ordres de grandeur en comparaison de la base où les contraintes n'ont pas été imposées. Par conséquent, cette stratégie est conservée par la suite.

Plusieurs perspectives sont alors envisageables. Une première piste serait d'ajouter des équations dans le système linéaire pour que les bases satisfassent certaines propriétés (*i.e.* des contraintes de symétrie). Cet ajout pourrait rendre la matrice moins sous-déterminée.

Une deuxième idée serait d'utiliser un préconditionneur pour le système ou de considérer d'autres équations à projeter pour construire le système. Pour ce dernier point, il est possible qu'en projetant une base de monômes orthogonalisée ou orthonormalisée (plutôt que considérer une base de monômes) conduise à un système mieux conditionné.

Enfin, l'utilisation d'un autre langage pour améliorer la précision numérique (*i.e.* considérer une précision supérieure à celle du double comme le quad en C++) serait utile pour construire le système linéaire et le résoudre.

Cependant, malgré le mauvais conditionnement du système linéaire, le point important réside dans la vérification des contraintes et des projections. Si tel est le cas, alors les solutions obtenues peuvent servir comme fonctions de base. Ce conditionnement devient uniquement un obstacle si l'on cherche à "contrôler" plus finement la solution retenue parmi l'infinité de solutions que présente ce système sous-déterminé.

Une synthèse du processus global d'obtention des fonctions r^i, r^{ij} est donné dans l'algorithme 4.

Algorithme 4 : Calcul des fonctions de base de Wachspress**Données** : k, m, q **Résultat** : w_{ij} et $w_{ij} \leftarrow$ coefficients des fonctions r^i, r^{ij} **si** q est inconnue **alors**| $q \leftarrow$ Appliquer l'algorithme GADJ de Dasgupta à l'ordre 1**fin****pour** chaque monôme $x^\alpha y^\beta \in P_k$ **faire**| Projection de $x^\alpha y^\beta$ dans la base de Wachspress basis avec les fonctions inconnues;

| Identifier les coefficients intervenant pour le monôme projeté à 1, sinon identifier à 0 pour les autres termes ;

fin**si** contraintes **alors**

| Ajout des contraintes géométriques dans le système linéaire

finDéterminer la matrice A associée aux coefficients recherchés pour le numérateur;Déterminer le membre de droite \vec{b} du système correspondant;Scinder les variables indépendantes et dépendantes \leftarrow matrice rref $M = [A, b]$;

Choisir une valeur arbitraire pour les variables indépendantes;

Inverser le système pour déduire les variables dépendantes;

Vérifier que la solution obtenue satisfait les projections et les contraintes géométriques.

2.2 Remarques sur le système linéaire

Preuve de l'inclusion des contraintes géométriques dans les équations de projection.

Montrons que l'ensemble des équations issues des projections Equation (2.8) sur les monômes contient les équations résultant des contraintes géométriques Equation (2.7).

Pour rappel, on considère un polygone convexe κ avec m côtés, pour un ordre k donné. Montrons alors que les Equations (2.7) sont incluses dans les Equations (2.8), **considérons un polynôme** $Q \in P_k$.

Décomposons Q dans la base des fonctions de Wachspress :

$$Q = \sum_{i=1}^m (w_i Q(a_i) + \sum_{j=1}^{k-1} w_{ij} Q(a_{ij}))$$

1) Prouvons que $w_i(a_l) = \delta_{i,l}$, $w_{i_j}(a_l) = 0$

Soit $l \in \llbracket 1; m \rrbracket$, on a :

$$\begin{cases} \forall i \in \llbracket 1; m \rrbracket \setminus \{l\}, w_i(a_l) = 0 \text{ car } p^i(a_l) = 0 \Rightarrow w_i(a_l)Q(a_l) = 0 \\ \forall i \in \llbracket 1; m \rrbracket, \forall j \in \llbracket 1, k-1 \rrbracket, w_{i_j}(a_l) = 0 \text{ car } p^{i_j}(a_l) = 0 \Rightarrow w_{i_j}(a_l)Q(a_{i_j}) = 0 \end{cases}$$

Si $Q(x, y) = 1$, ainsi : $Q(a_l) = w_l(a_l)Q(a_l) \Rightarrow w_l(a_l) = 1$

Finalement : $w_i(a_l) = \delta_{i,l}$, $w_{i_j}(a_l) = 0$

2) Prouvons que $w_i(a_{l_j}) = 0$, $w_{i-1}(a_{l_j}) = 0$, $w_{l_j}(a_{l_p}) = \delta_{j,p}$

Soient $i \in \llbracket 1; m \rrbracket, \forall j \in \llbracket 1, k-1 \rrbracket$, il s'ensuit :

$$\begin{cases} \forall i \in \llbracket 1; m \rrbracket \setminus \{l, l-1\}, w_i(a_{l_j})Q(a_{l_j}) = 0 \text{ car } p^i(a_{l_j}) = 0 \\ \forall i \in \llbracket 1; m \rrbracket \setminus \{l\}, w_{i_j}(a_{l_j})Q(a_{i_j}) = 0 \text{ car } p^{i_j}(a_{l_j}) = 0 \end{cases}$$

Ainsi :

$$Q(a_{l_j}) = w_{l-1}(a_{l_j})Q(a_{l-1}) + w_l(a_{l_j})Q(a_l) + \sum_{p=1}^{k-1} w_{l_p}(a_{l_j})Q(a_{l_p}) \quad (2.16)$$

Pour un polynôme 2D d'ordre k , il y a $N_{ddl} = \frac{(k+1)(k+2)}{2}$.

Si nous notons $G_l = \{a_l, a_{l-1}, \{a_{l_p}\}, p \in \llbracket 1, k-1 \rrbracket\}$, alors $\text{card}(G_l) = k+1$, vu que l'on a $N_{ddl} > \text{card}(G_l)$, une interpolation de Lagrange peut être réalisée, ainsi : $\forall a \in G_l, \exists Q_a \in P_k$:

$$\begin{cases} Q_a(a) = 1 \\ \forall a' \in G_l \setminus \{a\}, Q_a(a') = 0 \end{cases}$$

Par conséquent, si Q_a est utilisée (pour $a = a_{l_j}, a = a_{l_p}, \dots$) comme fonction Q dans l'Equation (2.16), on obtient alors : $\forall i, l \in \llbracket 1; m \rrbracket^2, \forall j, p \in \llbracket 1, k-1 \rrbracket^2$:

$$\begin{cases} w_{i_j}(a_{i_p}) = \delta_{j,p} \\ w_l(a_{i_j}) = 0, w_{l-1}(a_{i_j}) = 0 \end{cases}$$

Ainsi, les équations des contraintes géométriques sont incluses dans les équations de projection.

Sur la non existence de variables indépendantes aux ordres 1-2

Il est important de noter qu'aux ordres 1-2, il n'existe pas de variable libre.

- A l'ordre 1, $P = \text{Vect}(\{w_i\}_{i \in \llbracket 1; m \rrbracket})$ tels que $\forall i \in \llbracket 1; m \rrbracket w_i = c_i \frac{p^i}{q}$. Vu que $r^i = 1$ i.e. il n'y a pas d'inconnues.
- A l'ordre 2, $P = \text{Vect}(\{w_i, w_{i_1}\}_{i \in \llbracket 1; m \rrbracket})$ avec $\forall i \in \llbracket 1; m \rrbracket, w_i = c_i \frac{p^{i_1}}{q}$ et $w_{i_1} = c_{i_1} \frac{p^{i_1}}{q}$. Dans ce cas, il n'y a pas d'inconnues pour w_{i_1} vu que $r^{i_1} = 1$. Cependant, il est facile de prouver que la fonction r^i pour w_i est unique. En effet, les fonctions r^i vérifient les contraintes $w_i(a_{(i-1)_1}) = w_i(a_{i_1}) = 0$, étant donné qu'à l'ordre 2, r^i est une fonction polynomiale d'ordre 1, r^i correspond alors à l'équation de la droite passant par les points $a_{(i-1)_1}$ et a_{i_1} . Ainsi r^i est unique, de ce fait, il n'y a pas de variables indépendantes à l'ordre 2.

2.3 Transformation homothétique de la solution obtenue

Pour pouvoir calculer les fonctions de base pour un polygone convexe à une échelle quelconque, une transformation homothétique est réalisée. Pour cela, notons (x, y) les coordonnées associées au polygone κ . Pour obtenir ses coordonnées à une échelle différente, l'application linéaire suivante est définie : $x \mapsto l_x x$ et $y \mapsto l_y y$. Cela permet de générer le nouveau système de coordonnées : $(X, Y) = (l_x x, l_y y)$ associé au polygone $\tilde{\kappa}$ homothétique à κ . Détaillons ce processus pour une fonction polynomiale r avec pour coordonnées $(x, y) \in \kappa$:

$$r(x, y) = \sum_i c f_i x^{\alpha_i} y^{\beta_i} \quad (2.17)$$

Pour calculer cette fonction de base sur $\tilde{\kappa}$, nous introduisons la fonction R . Cette fonction se déduit de r en utilisant le nouveau système de coordonnées $(X, Y) = (l_x x, l_y y)$:

$$R(X, Y) = \sum_i \frac{c f_i}{l_x^{\alpha_i} l_y^{\beta_i}} X^{\alpha_i} Y^{\beta_i} \quad (2.18)$$

Par conséquent, si les bases de la fonction de Wachspress ont été calculées par notre méthode pour un κ donné, elles peuvent être déduites par cette transformation linéaire pour n'importe quelle polygone homothétique à κ . En particulier, dans le cadre d'un maillage hexagonal régulier, cela signifie que le calcul des fonctions de Wachspress peut être effectué une seule fois.

Comme discuté et illustré plus en détail dans la section 3, l'utilisation de cette transformation présente également un avantage en termes de stabilité numérique et de précision.

2.4 Détails d'implémentation

Le code pour générer les bases Wachspress est open source et peut être trouvé ici : <https://github.com/Dalab25/genWachspressBasisForPolygons>. La bibliothèque symbolique Python, sympy [74], a été utilisée⁶. Ainsi, le système d'inconnues *i.e.* les coefficients des polynômes 2D r^i et r^{ij} sont formellement représentés comme des objets `Symbol` et contenus dans un vecteur \vec{x} . Toutes les équations algébriques introduites dans la section 2 sont implémentées à l'aide de sympy afin d'établir le système linéaire. À cette fin, les fonctionnalités suivantes de sympy sont principalement utilisées :

- `sympy.Add.make_args` est utilisé pour stocker tous les termes devant les monômes dans un `dict` Python.
- `sympy.expand` est utile pour simplifier une expression, par exemple dans notre cas d'utilisation, il développe les polynômes factorisés en une somme de monômes.
- `sympy.Poly` transforme les objets sympy en un objet `sympy.poly`, d'où nous pouvons extraire les différents monômes.
- `sympy.fraction` renvoie le numérateur et le dénominateur d'une expression rationnelle.

Le pseudo-code Python dans Figure 2.4 est un extrait du script qui a été utilisé pour générer les coefficients. Celui-ci représente une implémentation de l'algorithme 4. Dans ce cadre, la fonction `getExpr` renvoie une expression formelle correspondant à la projection d'un monôme dans la base des fonctions de Wachspress. Cette projection s'effectue à l'aide des variables symboliques x et y . La fonction `getDictFromExpr` est la fonction pour transformer une expression sympy en un objet dictionnaire `dict` Python que nous pouvons ensuite manipuler pour construire le système linéaire. Les lignes 22 à 27 du code correspondent à des manipulations algébriques pour extraire les matrices associées aux variables dépendantes et indépendantes ainsi que les éléments correspondants réordonnés dans le vecteur \vec{b} . Cela permet alors de résoudre Equation (2.15) en fixant les variables indépendantes pour en déduire les variables dépendantes et donc obtenir les coefficients des fonctions r^i, r^{ij} .

Le processus de calcul de la matrice `rref` nécessite des précautions car des problèmes numériques peuvent apparaître à cause de l'opération de pivotement. En effet, cette opération est basée sur une tolérance qui modifie le rang de la matrice `rref` en fonction de sa valeur [44]. Dans un premier temps, la méthode `sympy.Matrix.rref()` a été utilisée à cet effet mais des problèmes numériques sont survenus en raison du manque de contrôle de la tolérance ; pour surmonter ce problème, une méthode inspirée de [93] a été utilisée.

6. Ce choix est purement arbitraire et d'autres librairies comme Mathematica auraient pu être utilisées.

```

1   degrees = [(i, j) for i in range(order+1) for j in range(order+1)
2             if i+j <= order]
3   for beta, gamma in degrees:
4       print(f"-----> Checking beta = {beta}, gamma = {gamma} ==> {x
5             **beta}*{y**gamma}")
6       monomial = lambda a, b: a**beta * b**gamma
7       # Get Sympy expression and the corresponding unknowns
8       # getExpr function sets up the WSF with the unknown
9       # and provides the unknown coefficients
10      expr, unknowns = getExpr(pitch, order, monomial)
11
12      # Simplifies an expression by developping the factorised
13      terms to monomials
14      expr = sympy.expand(expr)
15      # Returns a dict object with coefficients associated to each
16      monomial
17      dictCoeff = getDictFromExpr(expr, x, y)
18      # Set all the rational expressions on a common denominator
19      denominators = getDenominatorsForXiYj(dictCoeff)
20      dictXiYjTermsCommonDen = putAllNumeratorsOnCommonDenominator(
21          dictCoeff, denominators, verbose=0)
22      dictAllMonomialsProjected[(beta, gamma)] =
23          dictXiYjTermsCommonDen
24
25      A, B = setUpLinearSystemFromDictNew(dictAllMonomialsProjected,
26          unknowns, pitch)
27      M_rref1 = compute_M_rref(A, B, order, pitch, ...)
28      # Returns the positions of the independent variables in the
29      vector of unknowns
30      freePosVariables = convert_data_into_free_pos_variables(order,
31          pitch, nSides)
32
33      # Transforms M to matrix A_tild and a RHS vector b_tild
34      A_tild, b_tild = modify_M_to_Atild(order, pitch, nSides)
35      # Splits the matrix A_tild into two matrices
36      # One matrix for dependent variables A_d
37      # The other one for independent variables A_i
38      A_d, A_i = create_A_d_A_l(A_tild, order, pitch, nSides)
39      x_d = np.linalg.solve(A_d, b_tild-np.dot(np.transpose(A_i),x_i))

```

FIGURE 2.4 Extrait du code Python qui a été utilisé pour produire les fonctions de base.

3 Applications de l'algorithme de construction de base

L'algorithme de construction de bases est maintenant appliqué pour deux polygones convexes : l'hexagone régulier aux ordres 3-5 et le pentagone irrégulier aux ordres 3 et 4. Le système linéaire étant sous-déterminé pour ceux polygones, les solutions sont obtenues en utilisant la matrice M_{rref} et en fixant arbitrairement les variables indépendantes à zéro.

3.1 Application à un hexagone régulier

Dans cette section, les bases de Wachspress sont calculées des ordres 3 à 5 pour l'hexagone régulier. Le pas de l'hexagone p (*i.e.* la distance entre deux côtés parallèles quelconques), sauf spécification explicite, a été fixé à 1. À titre d'illustration, les Figures 2.5 et 2.6 représentent les fonctions de base associées aux degrés de liberté du bord inférieur de l'hexagone. Plus précisément, les Figures 2.5a, 2.5b, 2.6a et 2.6b montrent les courbes associées à w^1 , w_{1_1} , w_{1_2} et w_{1_3} à l'ordre 4, respectivement pour les nœuds d'ancrage $a_1, a_{1_1}, a_{1_2}, a_{1_3}$.

Les figures de la colonne de gauche représentent les courbes de niveau des fonctions de base sur l'hexagone régulier. On peut voir que ces fonctions sont C^∞ sur l'hexagone. Les tracés de la colonne de droite représentent la valeur de ces fonctions rationnelles sur les bords adjacents à a_1 . Les fonctions obtenues vérifient la propriété de Lagrange comme démontré par Wachspress. Cela conduit alors à des fonctions w_{1_i} égales à zéro sur les côtés opposés à leurs nœuds d'ancrage. Une remarque intéressante peut être faite pour les fonctions w_{1_1} et w_{1_3} , ces fonctions atteignent des valeurs supérieures à un avant d'atteindre l'unité sur leurs nœuds d'ancrage, un fait qui n'est généralement pas observé pour des fonctions de base polynomiales. En outre, ces figures montrent que les fonctions rationnelles ont des valeurs arbitraires à l'intérieur du domaine hexagonal, mais qu'elles décroissent comme une fonction polynomiale sur les bords.

Pour s'assurer que les bases sont correctement calculées, différents tests de vérification sont maintenant présentés. Premièrement, comme détaillé dans la section 2.4, la tolérance de pivotement dans le calcul de la matrice rref doit être choisie avec soin pour obtenir des résultats significatifs. Pour cela, une analyse paramétrique a été effectuée et il a été constaté qu'une valeur "optimale" (en termes d'erreurs discutées par la suite) peut être trouvée. L'étude s'est effectuée en considérant des valeurs suffisamment "élevées" (10^{-5}) puis à les diminuer jusqu'à ce que le rang de la matrice rref commence à augmenter. En pratique, une tolérance de 10^{-10} a été retenue pour les bases aux ordres 3 à 5 pour un pas de 1.

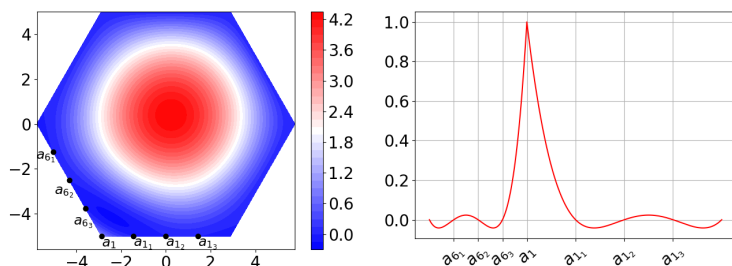
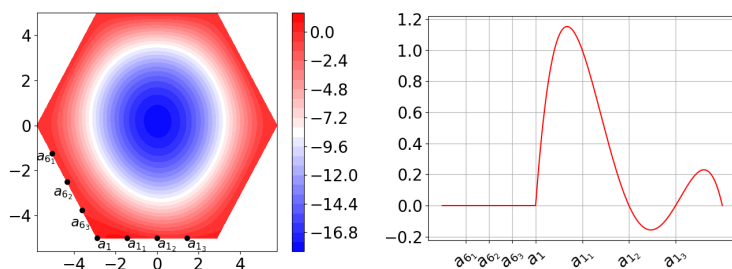
(a) Fonction w_1 (b) Fonction w_{1_1}

FIGURE 2.5 Illustrations des fonctions de base w_1, w_{1_1} à l'ordre 4. Pour les nœuds d'ancrage a_1, a_{1_1} respectivement, ces fonctions respectent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.

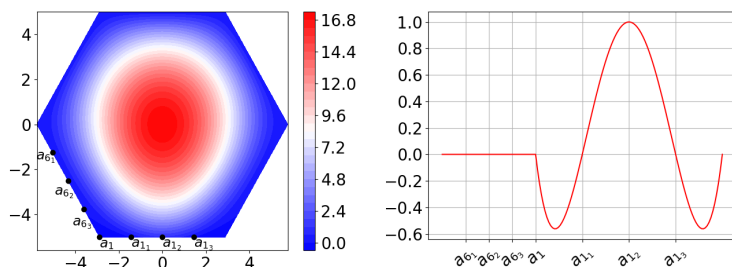
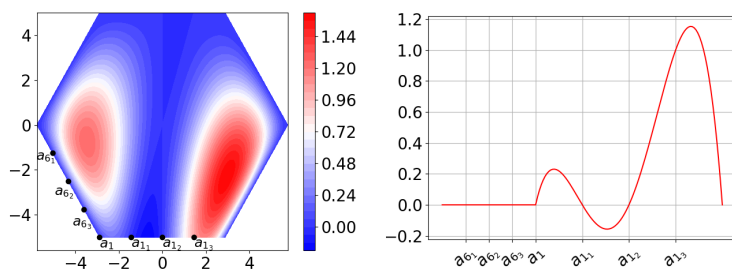
(a) Fonction w_{1_2} (b) Fonction w_{1_3}

FIGURE 2.6 Illustrations des fonctions de base w_{1_2}, w_{1_3} à l'ordre 4. Pour les nœuds d'ancrage a_{1_2}, a_{1_3} respectivement, ces fonctions présentent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.

Ensuite, nous nous assurons que les fonctions de base calculées respectent les propriétés décrites dans Equation (2.7) et Equation (2.8). Pour ce faire, les erreurs sur les contraintes géométriques notées ε_{w_i} , $\varepsilon_{w_{i_j}}$ et les erreurs de projection notées e_k ont été calculées *a posteriori*; ces quantités sont formellement définies dans Appendix 1 et Appendix 2 respectivement. On obtient alors $\varepsilon_{w_i} \leq 10^{-12}$ et $\varepsilon_{w_{i_j}} \leq 10^{-12}$ pour l'ordre 5, les erreurs étant plus faibles pour les ordres 3 et 4. Aussi, à l'ordre 3, une comparaison intéressante peut être faite entre notre solution particulière obtenue en fixant les variables indépendantes à zéro et la base proposée par Gout à l'ordre 3 dans [46] (détaillée en début de chapitre dans la section 1.1). Le choix des coefficients proposé par Gout conduit à des erreurs numériques autour de 10^{-16} , alors que les bases obtenues via notre processus numérique produisent des erreurs autour de 10^{-14} . En ce qui concerne l'erreur de projection e_k , celle-ci est toujours inférieure ou égale à 10^{-27} , et est donc insignifiante. Par conséquent, les monômes sont correctement projetés dans l'espace couvert par les fonctions de base obtenues $\{w_i, w_{i_j}\}$. Si l'on compare une nouvelle fois notre base à l'ordre 3 à celle de Gout, cette dernière génère une erreur de projection inférieure à la notre avec une erreur autour de 10^{-31} .

Ainsi, malgré le mauvais conditionnement du système linéaire comme expliqué dans la deuxième section, les erreurs numériques sont très faibles sur les contraintes et les projections. Par conséquent, les fonctions obtenues peuvent servir comme fonctions de base.

Pour tester la robustesse de la méthode, des calculs d'erreur sur les contraintes géométriques ainsi que sur les erreurs de projection ont été réalisés pour différents pas appartenant à l'intervalle $[1/1000, 1000]$. Ces erreurs sont reportées dans le Tableau 2.1 et ont été calculées pour des bases à l'ordre 3. Pour tout pas p , deux solutions de base différentes sont comparées : celle calculée directement sur l'hexagone au pas p et celle correspondant à une mise à l'échelle à partir de la solution obtenue pour un hexagone avec un pas de 1.

On peut remarquer que les erreurs liées aux contraintes géométriques sont presque insensibles à la modification du pas alors que l'erreur de projection augmente lorsque le pas augmente (cette croissance est même plus rapide que la surface de l'hexagone qui croît en $O(p^2)$). Cette dernière tendance est plus prononcée aux ordres 4-5 (non montrée ici) mais dans la plupart des cas, les grandes valeurs de pas ne présentent aucun intérêt pour les applications pratiques et cette augmentation de la projection n'est pas nuisible. Il est également très intéressant de noter que la solution mise à l'échelle est plus robuste que le calcul direct pour un pas p donné car le choix de la tolérance pour l'opération de pivotement pour le calcul de la matrice rref dépend de p . De plus, pour les petits pas, aucune tolérance adéquate n'a été trouvée. Il est donc plus simple de calculer une solution et de procéder ensuite à une mise à l'échelle plutôt que de calculer une solution pour chaque pas, de vérifier

si le nombre de variables indépendantes n'a pas changé et, le cas échéant, d'adapter le paramètre de tolérance en conséquence.

TABLEAU 2.1 Erreurs associées aux différentes bases calculées à l'ordre 3 pour certains hexagones de pas p .

p	e_k		$\max(\epsilon_{w_i}, \epsilon_{w_{i_1}}, \epsilon_{w_{i_2}})$	
	Solution directe	Solution mise à l'échelle	Solution directe	Solution mise à l'échelle
1/1000	NA *	10^{-55}	NA *	10^{-13}
1/100	NA *	10^{-47}	NA *	10^{-13}
1/10	10^{-37}	10^{-39}	10^{-13}	10^{-13}
1	10^{-31}	10^{-31}	10^{-13}	10^{-13}
10	10^{-23}	10^{-24}	10^{-14}	10^{-13}
100	10^{-15}	10^{-15}	10^{-14}	10^{-13}
1000	$10^{-7} \dagger$	10^{-7}	$10^{-13} \dagger$	10^{-13}

* : résultat "non disponible" car aucune tolérance pour l'opération de pivotement (même en considérant une très petite tolérance $< 10^{-16}$) ne donne un rang de matrice correct.

† : la tolérance pour l'opération de pivotement pour le calcul de la matrice rref doit être augmentée de 10^{-10} à 10^{-6} pour obtenir le rang correct de la matrice.

Enfin, quelques remarques intéressantes peuvent être ajoutées concernant les variables indépendantes. En effet, aux ordres 3-4-5, il y a respectivement 8-27-54 variables indépendantes. De plus, ces variables sont toujours associées à certaines fonctions r^{ij} et plus particulièrement, pour un ordre donné, il s'agira toujours du même type de coefficients pour chaque r^{ij} , ceci est résumé dans les Tableaux 2.2 et 2.3. Egalement, les variables indépendantes sont les coefficients devant les monômes de degré inférieur ou égal à $\deg(r^{ij}) - 1$.

TABLEAU 2.2 Fonctions présentant des variables indépendantes pour l'hexagone régulier

Ordre	Fonctions r^{ij} avec des variables indépendantes
3	$r^{62}, r^{52}, r^{42}, r^{32}, r^{22}, r^{12}, r^{61}, r^{51}$
4	$r^{63}, r^{53}, r^{43}, r^{33}, r^{23}, r^{13}, r^{62}, r^{52}, r^{42}$
5	$r^{64}, r^{54}, r^{44}, r^{34}, r^{24}, r^{14}, r^{63}, r^{53}, r^{43}$

TABLEAU 2.3 Variables indépendantes associées aux fonctions r^{ij} pour l'hexagone régulier

Ordre	forme des fonctions r^{ij}	variables indépendantes
3	$ax + by + c$	c
4	$ax^2 + by^2 + cxy + dx + ey + f$	d, e, f
5	$ax^3 + by^3 + cx^2y + dxy^2 + ex^2 + fy^2 + gxy + hx + iy + j$	e, f, g, h, i, j

3.2 Application au pentagone irrégulier

L'algorithme de création de bases de Wachspress a également été appliqué au cas du pentagone irrégulier que l'on peut voir sur la Figure 2.7.

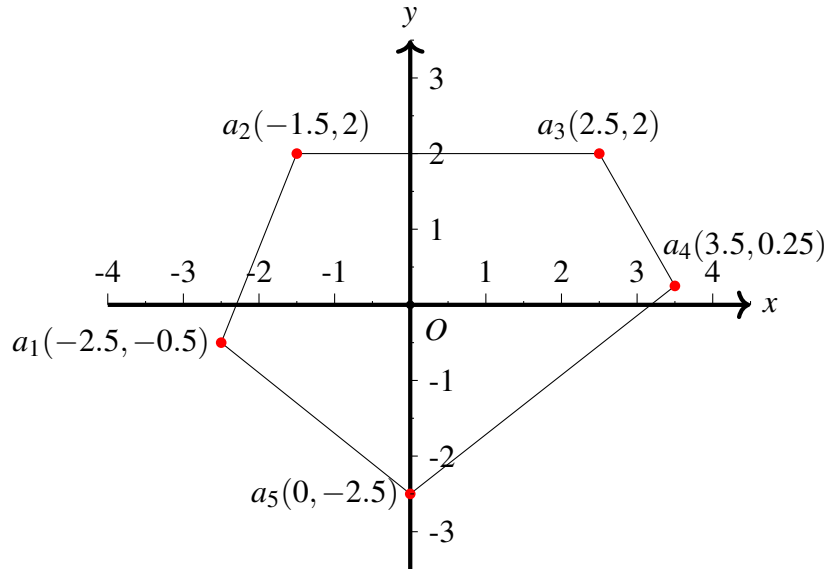


FIGURE 2.7 Le pentagone irrégulier considéré pour notre étude.

Contrairement au cas de l'hexagone régulier, le calcul de la fonction adjointe q est moins simple dans le cas d'un pentagone irrégulier car il ne s'agit plus de l'équation d'un cercle. Comme expliqué au début de ce chapitre, il aurait été possible d'utiliser l'algorithme général GADJ de Dasgupta dans [28]. Néanmoins, dans le cas du pentagone, nous pouvons utiliser le fait que $q \in P^2$ tel que q est une fonction de la forme $q(x, y) = x^2 + a_1y^2 + a_2xy + a_3x + a_4y + a_5$. De plus, l'adjoint passe par les cinq points d'intersection externe b_i de coordonnées (b_{ix}, b_{iy}) i.e. $q(b_i) = 0$, alors le système linéaire suivant de taille 5×5 peut être obtenu :

$$\begin{cases} a_1b_{1y}^2 + a_2b_{1x}b_{1y} + a_3b_{1x} + a_4b_{1y} + a_5 = -b_{1x}^2 \\ \vdots \\ a_1b_{5y}^2 + a_2b_{5x}b_{5y} + a_3b_{5x} + a_4b_{5y} + a_5 = -b_{5x}^2 \end{cases}$$

Sa résolution conduit à l'obtention des coefficients de l'adjoint q . Cette fonction q est illustrée sur la Figure 2.8.

Connaissant l'adjoint, la même méthode que dans le cas de l'hexagone a ensuite été appliquée pour calculer les fonctions de base associées à ce polygone aux ordres 3 et 4. Les

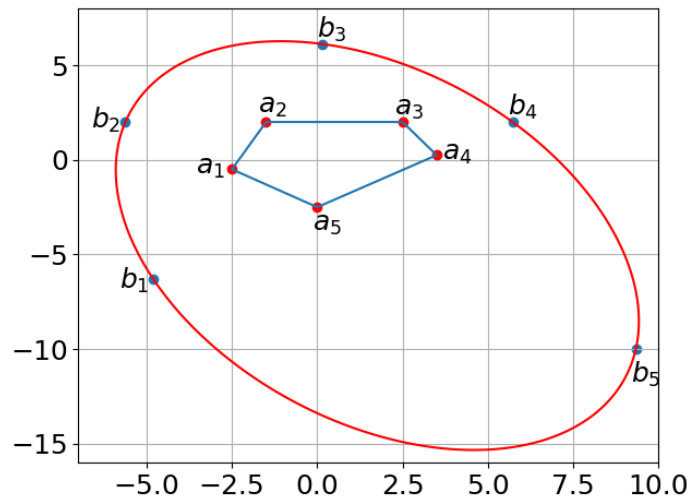


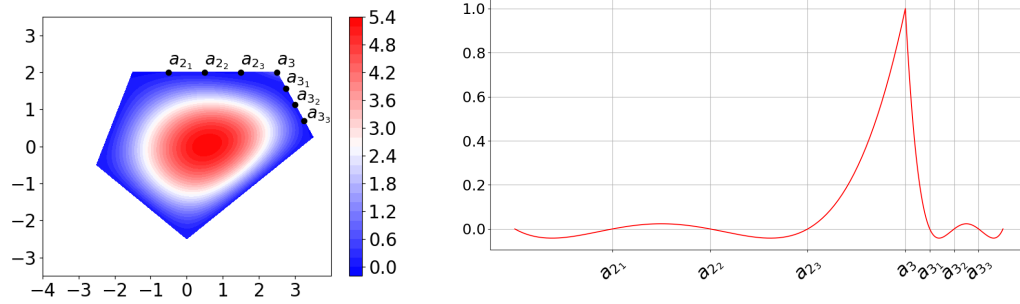
FIGURE 2.8 Pentagone irrégulier considéré dans ce travail. Les points a_i représentent les sommets du polygone. Les points b_i représentent les EIP qui permettent de définir l'adjoint q du pentagone irrégulier représentée ici par une ellipse en rouge.

Figure 2.9 et Figure 2.10 représentent les fonctions de base obtenues avec les variables libres fixées à 0 à l'ordre 4.

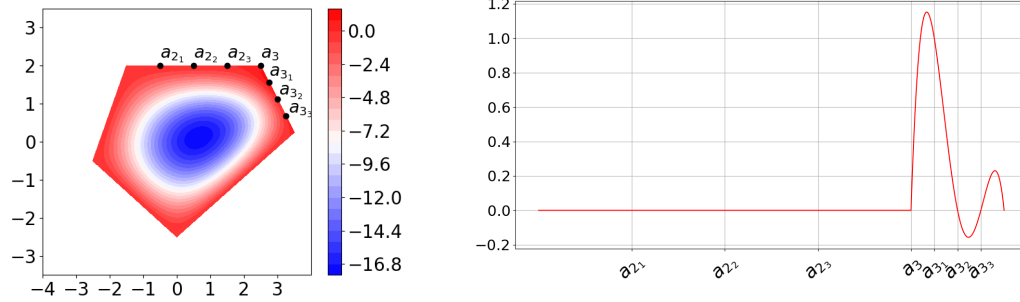
Pour chaque base, comme dans le cas de l'hexagone, les erreurs sur les contraintes géométriques et la projection ont été calculées et se sont avérées être du même ordre de grandeur que dans le cas de l'hexagone régulier. Par conséquent, les contraintes géométriques sont vérifiées et la base d'ordre k couvre l'espace des polynômes d'ordre k .

D'autres remarques peuvent être ajoutées sur les variables libres. De la même manière que dans le cas de l'hexagone régulier, ces dernières sont associées aux fonctions r^{ij} aux ordres 3-4. Plus particulièrement, contrairement au cas de l'hexagone, ces variables interviennent exclusivement pour les fonctions r^{i2} à l'ordre 3 et r^{i3} à l'ordre 4. Aussi, semblablement à l'hexagone, les variables indépendantes correspondent aux coefficients devant les monômes de degré inférieur ou égal à $\deg(r^{ij}) - 1$. Les Tableau 2.4 et Tableau 2.5 résument ces résultats.

Maintenant que nous avons appliqué notre méthode de génération de bases rationnelles à deux polygones convexes, nous allons présenter l'autre base utilisée pour réaliser des comparaisons par la suite.



(a) Fonction w_3



(b) Fonction w_3^1

FIGURE 2.9 Illustrations des fonctions de base à l'ordre 4 : w_3, w_3^1 pour les noeuds d'ancrage à a_3, a_{3_1} respectivement. Ces fonctions présentent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.

TABLEAU 2.4 Fonctions présentant des variables indépendantes pour le pentagone irrégulier

Ordre	Fonctions r^{ij} avec des variables indépendantes
3	$r^{5_2}, r^{4_2}, r^{3_2}, r^{2_2}, r^{1_2}$
4	$r^{5_3}, r^{4_3}, r^{3_3}, r^{2_3}, r^{1_3}$

TABLEAU 2.5 Variables indépendantes associées aux fonctions r^{ij} pour le pentagone irrégulier

Ordre	forme des fonctions r^{ij}	variables indépendantes
3	$ax + by + c$	c
4	$ax^2 + by^2 + cxy + dx + ey + f$	d, e, f

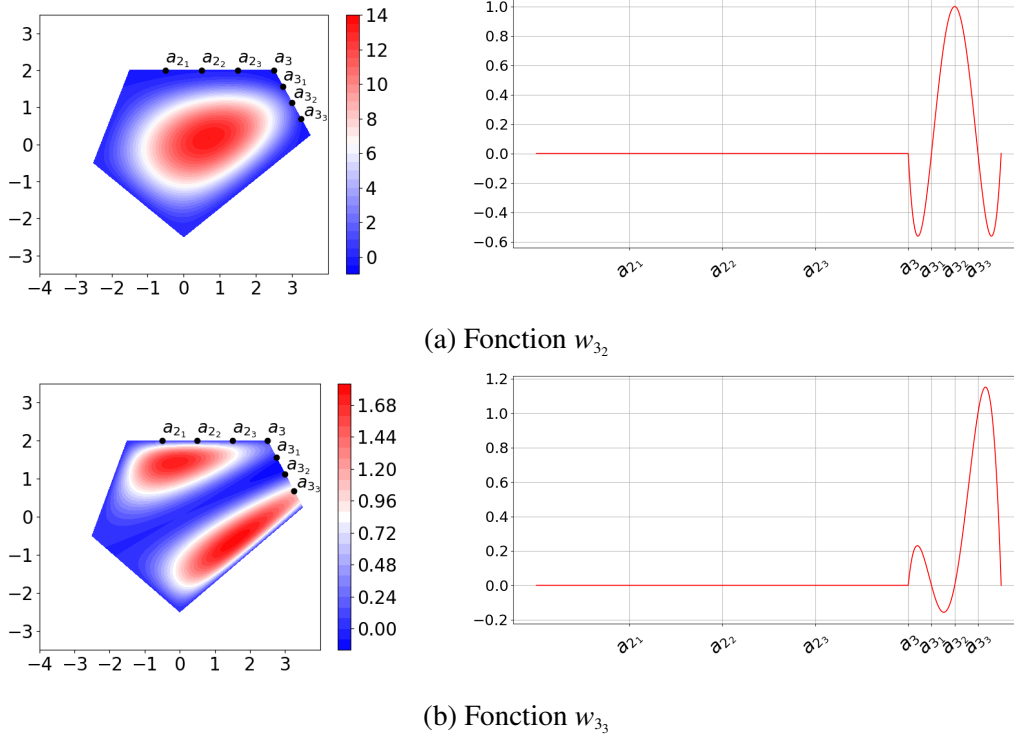


FIGURE 2.10 Illustrations des fonctions de base à l'ordre 4 : w_{3_2}, w_{3_3} pour les noeuds d'ancrage à a_{3_2}, a_{3_3} respectivement. Ces fonctions présentent la propriété de Lagrange attendue, comme l'illustrent les figures de la colonne de droite.

4 Développement de la base de monômes orthogonalisée

Ces bases de monômes orthogonalisées sont généralement utilisées dans un contexte DG sur des maillages polygonaux [57, 58]. Ces bases présentent l'avantage d'être pratiques à implémenter. Toutefois, précisons qu'elles ne peuvent être utilisées pour la méthode des éléments finis continus car elles ne permettent pas de garantir la continuité aux interfaces entre les différentes mailles.

Ici, nous allons détailler ces bases et plus particulièrement le processus d'orthogonalisation. Pour cela, nous reprenons les notations introduites pour définir un élément fini de Wachspress, *i.e.* la base de monômes orthogonalisée définit l'élément fini suivant (κ, P, Σ) où :

- κ : une maille correspondant à polygone convexe.
- P : espace d'approximation contenant $N_k = \binom{k+2}{k}$ fonctions de base définies sur κ .
- Σ : ensemble de formes linéaires sur P , aussi appelé ensemble des degrés de liberté.

A un ordre k , on définit P en orthogonalisant l'espace \tilde{P}_k suivant :

$$\tilde{P}_k = \text{Vect}(\{\forall i \in \llbracket 0, N_k \rrbracket, \mathbf{v}_i = \left(\frac{x-x_e}{p}\right)^{\alpha_i} \left(\frac{y-y_e}{p}\right)^{\beta_i}, \text{ avec } (\alpha_i, \beta_i) \in C_k\}) \quad (2.19)$$

où p est le pas de l'hexagone et (x_e, y_e) le barycentre de la maille κ . En pratique, nous considérons un maillage régulier donc chaque élément se résume en une translation de l'élément de référence. Par conséquent, la transformation géométrique permettant de se ramener à un élément de référence fait intervenir une matrice Jacobienne égale à la matrice identité.

Aussi, pour une fonction f donnée, l'espace Σ s'écrit :

$$\Sigma = \{\forall i \in \llbracket 0, N_k \rrbracket, \sigma_i(f) = \frac{(f, \mathbf{v}_i)_{L^2(\kappa)}}{\|\mathbf{v}_i\|_{L^2(\kappa)}^2}\} \quad (2.20)$$

En effet, si l'on considère la projection d'une fonction f dans P , cette dernière s'écrit :

$$\begin{aligned} f &= \sum_{i=0}^{N_k} f_i \mathbf{v}_i \\ \Rightarrow (f, \mathbf{v}_j)_{L^2(\kappa)} &= \sum_{i=0}^{N_k} f_i \langle \mathbf{v}_i, \mathbf{v}_j \rangle_{L^2(\kappa)} \\ \text{or } \sum_{i=0}^{N_k} f_i \langle \mathbf{v}_i, \mathbf{v}_j \rangle_{L^2(\kappa)} &= f_j \|\mathbf{v}_j\|_{L^2(\kappa)}^2 \text{ puisque la base est orthogonale} \\ \Leftrightarrow f_j &= \frac{(f, \mathbf{v}_j)_{L^2(\kappa)}}{\|\mathbf{v}_j\|_{L^2(\kappa)}^2} \end{aligned} \quad (2.21)$$

Dans ce cadre, on peut montrer que Σ est P -unisolvent. En effet, on a bien $\dim(\Sigma) = \dim(P)$ et $\sigma_i(\mathbf{v}_j) = \delta_{i,j}$.

Par ailleurs, il est important de noter que l'approximation d'une fonction continue par une méthode DG sur \mathcal{D} par une base de monômes conduit à un saut entre deux mailles tendant vers 0.

En pratique, la base de fonctions obtenue par P a été orthogonalisée à l'aide de l'algorithme de Gram-Schmidt [11] pour le produit scalaire $(\cdot, \cdot)_{L^2(\kappa)}$. Cette orthogonalisation conduit à des matrices de masse diagonales et améliore le conditionnement de ces dernières

en comparaison d'une base de monômes. Cependant, obtenir des matrices de masse diagonales n'est pas une nécessité pour réaliser des calculs de transport. En effet, l'opérateur de transport ne se résume pas seulement en une matrice de masse et également, pour notre discrétisation, nous résolvons des problèmes locaux de petites dimensions (de dimension $N(k)$). Par conséquent, considérer une matrice pleine ou creuse n'aura que peu d'incidence.

Pour réaliser la procédure d'orthogonalisation, nous utilisons une nouvelle fois le calcul symbolique avec la librairie `sympy` de python. Cet outil permet d'appliquer l'algorithme de Gram-Schmidt en transformant chaque fonction de base monomiale en une expression `sympy` qui est ensuite orthogonalisée. Une fois cette étape réalisée, à l'aide de la fonctionnalité `sympy.lambdify`, l'expression "orthogonalisée" devient alors une fonction que l'on ajoute dans une liste correspondant à la nouvelle base de fonctions orthogonalisée. Les détails pour les calculs des normes et des produits scalaires $L^2(\kappa)$ seront donnés dans le chapitre suivant.

5 Conclusion et perspectives

Dans ce chapitre, une méthode originale pour générer les fonctions de Wachspress a été développée et appliquée à deux polygones convexes, l'hexagone régulier et le pentagone irrégulier. Celle-ci se base sur les propriétés que doivent vérifier les fonctions de base et utilise le calcul symbolique pour construire un système linéaire dont les inconnues sont les coefficients des fonctions r^i, r^{ij} . Un premier résultat intéressant a consisté en une démonstration théorique pour montrer que les équations de projections incluent les équations des contraintes géométriques, ce qui a bien été vérifié numériquement. Aussi, un autre résultat clé est l'infinité de bases de Wachspress possibles. Cela provient du système linéaire obtenu qui est sous-déterminé. Ainsi, en considérant la matrice rref de ce système, des variables dépendantes et indépendantes peuvent être déduites et en fonction des choix pour ces dernières, différentes solutions peuvent être obtenues. Gout a notamment fait des choix pour ses fonctions de base à l'ordre 3. Ces choix ont une incidence sur les erreurs d'approximation, c'est notamment ce qui a pu être observé lors d'une comparaison sur les erreurs de projections et de contraintes entre la base de Gout et la notre à l'ordre 3 pour l'hexagone régulier. Egalement, une base de monômes orthogonalisée a été développée pour être comparée par la suite aux bases de fonctions de Wachspress.

En perspective, un point important serait l'amélioration du conditionnement du système linéaire, plus particulièrement le conditionnement de la matrice A_d intervenant dans la matrice échelonnée réduite. Pour cela, une première idée repose sur l'utilisation d'un préconditionneur combinée à une précision accrue de la représentation flottante⁷

Une autre perspective serait de développer des algorithmes pour imposer certaines propriétés aux fonctions de bases à partir des variables indépendantes. Plus particulièrement, les fonctions de base pourraient vérifier certaines propriétés de symétrie et/ou de rotations inhérentes au polygone considéré. Aussi, au lieu de rajouter l'ensemble de ces contraintes dans le système linéaire (ce qui pourrait le rendre sur-déterminé), une alternative consisterait à le reformuler en un problème d'optimisation. Plus particulièrement, effectuer la minimisation d'une fonctionnelle (par exemple appliquer un opérateur de rotation à l'ensemble des fonctions de base moins sa projection dans la base) sous contraintes (propriété de Lagrange, projection des monômes).

7. La précision des nombres flottants en python est en double. On pourrait utiliser une représentation flottante plus précise pour les opérations intervenant dans l'algorithme d'évaluation de la base.

Chapitre 3

Résolution d'équations aux dérivées partielles sur des maillages hexagonaux

Suite aux développements du chapitre précédent, nous nous assurons que les bases produites sont adaptées à la discrétisation d'équations aux dérivées partielles sur des maillages hexagonaux. Pour cela, nous utilisons la méthode des solutions manufacturées [90] pour vérifier que les ordres de convergence obtenus coïncident avec ceux décrits par des estimateurs d'erreur *a priori* [68, 89]. En ce sens, des fonctions $C^\infty(\mathcal{D})$ n'appartenant pas à l'espace d'approximation sont utilisées.

Le premier cas d'application de cette méthode est l'équation de Poisson discrétisée par une méthode d'éléments finis continus avec les fonctions de Wachspress. Cette résolution sert de cas introductif avant de traiter l'équation de transport.

Pour cette dernière, celle-ci est discrétisée par une méthode S_n -DG comme présentée au chapitre 1. Nous nous assurons que les bases de Wachspress et les bases de monômes orthogonalisés produisent les ordres de convergence attendus.

Suite à ces analyses, le deuxième grand objectif consiste à comparer les bases de monômes orthogonalisées et des bases de Wachspress pour la résolution de l'équation de transport. Pour cela, des études supplémentaires sont réalisées via la méthode des solutions manufacturées. En particulier, des solutions faiblement régulières appartenant au même espace que la solution de transport sont utilisées comme solutions manufacturées.

Ces résultats fournissent des premiers éléments de comparaison avant de réaliser des calculs sur des cœurs de réacteurs. Pour ces calculs, afin de répondre à l'objectif initial de

cette thèse, des analyses sont effectuées sur le k_{eff} et sur les taux d'absorption au travers de deux benchmarks. Notons que l'ensemble de ces développements s'est déroulé dans une maquette (mini-application) développée en python.

1 Intégration numérique sur l'hexagone régulier

L'obtention de solutions approchées d'équations aux dérivées partielles par la méthode des éléments finis passe par la résolution d'un système linéaire obtenu à partir de la forme variationnelle discrète de l'équation considérée. Cette formulation fait intervenir des intégrales sur chaque maille du domaine (ici des hexagones réguliers), il est alors nécessaire de pouvoir les calculer numériquement. Deux stratégies sont possibles : l'utilisation de quadratures spécialement construites pour l'hexagone régulier [77], ou le partitionnement de ce polygone en triangles. Ici, nous optons pour la deuxième idée en découpant un hexagone régulier en six triangles par simplicité. Notons que pour notre application, le maillage est constitué d'hexagones réguliers avec des propriétés constantes par mailles. Par conséquent, une seule évaluation des matrices mises en jeu sur un hexagone est nécessaire à l'assemblage du système linéaire complet. Ainsi, le coût de l'intégration numérique est très faible. Pour cette raison, nous ne recherchons pas à optimiser la quadrature utilisée dans ces travaux.

1.1 Partitionnement et transformation de Duffy

Pour effectuer ce partitionnement d'un hexagone en six triangles, nous définissons d'abord un triangle de référence noté ABC comme illustré sur la Figure 3.1.

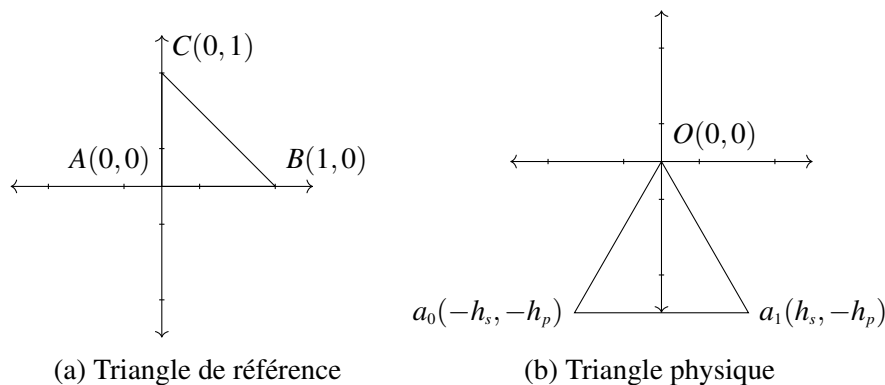


FIGURE 3.1 Transformation \mathcal{T} de triangle de référence vers le triangle physique du système.

En considérant un triangle général PQR de coordonnées (x_p, y_p) , (x_q, y_q) et (x_r, y_r) , la transformation \mathcal{T} des coordonnées (ξ, η) de ABC à (x, y) dans PQR s'exprime comme :

$$\begin{aligned} x &= x_p + (x_q - x_p)\xi + (x_r - x_p)\eta \\ y &= y_p + (y_q - y_p)\xi + (y_r - y_p)\eta \end{aligned}$$

Par conséquent, la matrice Jacobienne J pour \mathcal{T} s'écrit de la manière suivante :

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix} = \begin{pmatrix} x_q - x_p & x_r - x_p \\ y_q - y_p & y_r - y_p \end{pmatrix} \quad (3.1)$$

Si l'on considère que le triangle équilatéral physique est Oa_0a_1 (ce qui représente $\frac{1}{6}$ de l'hexagone régulier) avec des coordonnées (x, y) , J est alors égale à :

$\begin{pmatrix} -h_s & -h_p \\ h_s & -h_p \end{pmatrix}$ où h_s et h_p sont respectivement la moitié d'une arête et du pas de l'hexagone. Également, le déterminant de J noté $|J|$ est donné par $2h_s h_p$, et il en est de même pour les six triangles de l'hexagone régulier.

A partir de cette décomposition de l'hexagone régulier en six triangles, l'intégrale d'une fonction f sur une maille hexagonale κ s'écrit :

$$\int_{\kappa} dx dy f(x, y) = \sum_{i=1}^6 \int_{T_i} dx dy f(x, y)$$

Où les T_i sont les six triangles équilatéraux de l'hexagone régulier. Chaque intégrale sur T_i peut se ramener à l'élément de référence en effectuant la transformation géométrique suivante :

$$\int_{T_i} dx dy f(x, y) = \int_{ABC} d\xi d\eta f(\xi, \eta) |J| \quad (3.2)$$

Néanmoins, les fonctions peuvent rarement être intégrées exactement sur un triangle. Ainsi, pour pouvoir les calculer numériquement, nous devons donc recourir à des quadratures numériques. Une démarche possible est celle expliquée par Rathod [85]. Le point de départ est l'intégration de la fonction sur le triangle de référence comme décrit par l'Equation (3.2) (sans prendre en compte le terme Jacobien pour l'instant). Ainsi, cette équation peut être

approchée par une quadrature numérique sous la forme :

$$\int_{ABC} d\xi d\eta f(\xi, \eta) = \int_0^1 d\xi \int_0^{1-\xi} d\eta f(\xi, \eta) = \sum_{n=1}^N \omega_n f(\xi_n, \eta_n) \quad (3.3)$$

où ω_n sont les poids associés aux nœuds de quadrature (ξ_n, η_n) .

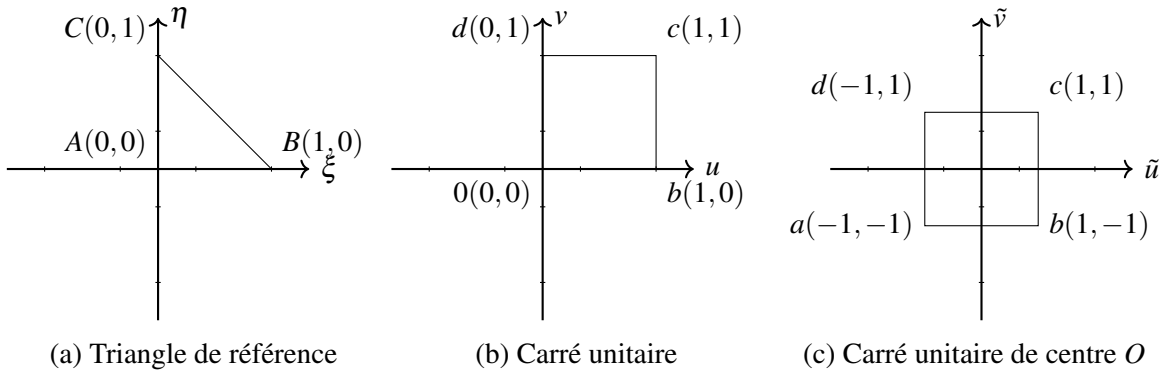


FIGURE 3.2 Transformation du triangle de référence vers le carré unitaire de centre O .

Pour utiliser une double quadrature de Gauss-Legendre sur un triangle, une transformation de Duffy [77] est opérée pour formuler cette intégrale sur le carré unitaire donné par $\{(u, v); 0 \leq u, v \leq 1\}$. Pour cela, les substitutions suivantes sont effectuées (les différentes transformations de cette partie sont résumées dans la Figure 3.2) :

$$\begin{aligned} \xi &= uv \\ \eta &= u(1-v) \\ d\xi d\eta &= ududv \end{aligned} \quad (3.4)$$

en injectant l'Equation (3.4) dans l'Equation (3.3), on obtient alors :

$$\int_0^1 d\xi \int_0^{1-\xi} d\eta f(\xi, \eta) = \int_0^1 du \int_0^1 dv f(uv, u(1-v))u \quad (3.5)$$

cette intégrale peut être exprimée sur le carré unitaire centré sur l'origine $\{(\tilde{u}, \tilde{v}); -1 \leq \tilde{u}, \tilde{v} \leq 1\}$ en utilisant les transformations suivantes pour u et v .

$$\begin{aligned}
u &= \frac{1 + \tilde{u}}{2} \\
v &= \frac{1 + \tilde{v}}{2} \\
dudv &= \frac{1}{4} d\tilde{u}d\tilde{v}
\end{aligned} \tag{3.6}$$

L'Equation (3.6) est substituée dans l'Equation (3.5) et l'expression finale s'écrit alors :

$$\int_0^1 d\xi \int_0^{1-\xi} d\eta f(\xi, \eta) = \int_0^1 du \int_0^1 dv f(uv, u(1-v))u \tag{3.7}$$

$$= \int_{-1}^1 d\tilde{u} \int_{-1}^1 d\tilde{v} f\left(\frac{(1+\tilde{u})(1+\tilde{v})}{4}, \frac{(1+\tilde{u})(1-\tilde{v})}{4}\right) \frac{(1+\tilde{u})}{8} \tag{3.8}$$

Ainsi, les quadratures standards de Gauss-Legendre peuvent être utilisées pour les deux variables et la quadrature numérique pour un ordre m donné avec m^2 points s'exprime :

$$\int_0^1 d\xi \int_0^{1-\xi} d\eta f(\xi, \eta) = \int_{-1}^1 d\tilde{u} \int_{-1}^1 d\tilde{v} \frac{(1+\tilde{u})}{8} f(\xi(\tilde{u}, \tilde{v}), \eta(\tilde{u}, \tilde{v})) \tag{3.9}$$

$$= \sum_{i=1}^m \sum_{j=1}^m \frac{(1+\tilde{u}_i)}{8} \omega_i \omega_j f(\xi(\tilde{u}_i, \tilde{v}_j), \eta(\tilde{u}_i, \tilde{v}_j)) \tag{3.10}$$

$$= \sum_{k=1}^{N=m \times m} \omega_k f(\xi_k, \eta_k) \tag{3.11}$$

avec $(\tilde{u}_i, \tilde{v}_j)$ le produit des points de Gauss-Legendre avec les poids associés ω_i et ω_j , et donc :

$$\xi_k = \frac{(1+\tilde{u}_i)(1+\tilde{v}_j)}{4} \quad \eta_k = \frac{(1+\tilde{u}_i)(1-\tilde{v}_j)}{4} \quad \omega_k = \frac{(1+\tilde{u}_i)}{8} \omega_i \omega_j$$

Les Figures C.1a et C.1b dans l'Annexe C représentent les nœuds des quadratures.

1.2 Intégration de fonctions sur le bord des hexagones

Pour pouvoir intégrer les fonctions de base sur le bord d'une cellule donnée, une quadrature de Gauss-Legendre est utilisée sur l'intervalle de référence $I_0 = [-1, 1]$. Une transformation géométrique \mathcal{T}_x est alors définie pour chaque arête d'équation y . Ainsi, on a $\mathcal{T}_x(x) = x(\xi_k)$ avec $\xi_k \in I_0, \Rightarrow y(x) = y(x(\xi_k))$. Par conséquent, en connaissant le numéro de l'arête, la coordonnée $x(\xi_k)$ peut être déduite afin de calculer la coordonnée y .

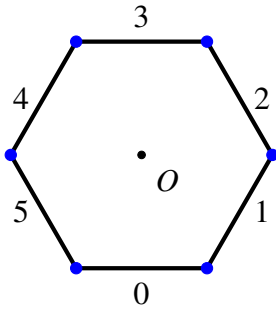


FIGURE 3.3 Numérotation des arêtes.

TABLEAU 3.1 Transformation géométrique de l'intervalle de référence I_0 vers les arêtes hexagonales physiques ($\xi \in I_0$). La numérotation locale des arêtes est donnée par la Figure 1.2.

Numéro de l'arête	\mathcal{T}_x	Equation de l'arête : $y(x)$
0	$h_s \xi_k$	$-h_p$
1	$\frac{h_s}{2}(\xi_k + 3)$	$\sqrt{3}x - 2h_p$
2	$\frac{h_s}{2}(\xi_k + 3)$	$-\sqrt{3}x + 2h_p$
3	$h_s \xi_k$	h_p
4	$\frac{h_s}{2}(\xi_k - 3)$	$\sqrt{3}x + 2h_p$
5	$\frac{h_s}{2}(\xi_k - 3)$	$-\sqrt{3}x - 2h_p$

Les poids de la quadrature Gauss-Legendre quadrature sont ajustés par un facteur en h_s .

Ainsi, l'intégrale d'une fonction f sur un bord b de la cellule κ s'écrit :

$$\int_b f dS = \sum_{\xi_k \in I_0} \omega_k f(x(\xi_k), y(x)) \quad (3.12)$$

1.3 Vérifications numériques

Nous devons nous assurer que l'erreur d'intégration est négligeable par rapport à l'erreur de discrétisation. Par conséquent, nous nous fixons un seuil arbitraire correspondant à une erreur d'intégration (ou à un calcul de résidu) inférieure à 10^{-15} . Ainsi, nous allons intégrer différentes fonctions (qui seront utilisées par la suite) afin d'en déduire l'ordre de quadrature respectant notre critère.

Premièrement, des vérifications sont réalisées en intégrant des fonctions polynomiales. Pour rappel, les quadratures de Gauss-Legendre d'ordre k permettent d'intégrer exactement des polynômes d'ordre $2k - 1$, dans ce cadre, la fonction $f(x, y) = x^2 + y^2$ peut être intégrée

exactement pour une quadrature d'ordre 2. Ce résultat a été confirmé en comparant l'intégrale numérique à son expression analytique *i.e.* $\int_{\kappa} x^2 + y^2 dV = 10h_s^3 h_p$. On remarquera bien que pour une quadrature d'ordre 1, la fonction $f(x, y) = x^2 + y^2$ n'est pas intégrée exactement comme l'illustre le Tableau 3.2.

Également, dans l'objectif d'effectuer des tests avec la méthode des solutions manufacturées avec une fonction $f(x, y) = \sin(x) \sin(y)$, il est nécessaire de regarder l'ordre de la quadrature pour lequel l'erreur d'intégration est inférieure à 10^{-15} . De par les symétries de l'hexagone régulier, les intégrales des fonctions impaires sont nulles, donc $\int_{\kappa} \sin(x) \sin(y) dV = 0$. Numériquement, $\int_{\kappa} \sin(x) \sin(y) dV \leq 10^{-15}$ pour des ordres de quadratures supérieurs ou égaux à cinq.

Enfin, le point le plus important consiste à déterminer le nombre de points de quadrature suffisant pour intégrer correctement les fonctions de Wachspress. Pour cela, nous introduisons :

$$i_m^{i,j} = \int_{\kappa} w_i w_j dV = \sum_{l=1}^{N=6 \times m \times m} \omega_l w_i(\xi_l, \eta_l) w_j(\xi_l, \eta_l) \quad (3.13)$$

avec m l'ordre de la quadrature.

Le résidu suivant est étudié $r_m^{i,j} = |i_m^{i,j} - i_{m-1}^{i,j}|$, de sorte à obtenir $e_m^{i,j} < 10^{-15}$. Ce résultat est atteint pour un ordre de quadrature d'ordre 15. Le Tableau 3.2 donne quelques valeurs numériques en fonction de l'ordre de la quadrature.

TABLEAU 3.2 Erreurs d'intégration numérique sur une cellule hexagonale.

Ordre m	Nombre de points	$e_m = \int_{\kappa} x^2 + y^2 - 10h_s^3 h_p $	$e_m = \int_{\kappa} \sin(x) \sin(y) $	$r_m^{1,5} = i_m^{1,5} - i_{m-1}^{1,5} $
1	6	6.0×10^{-3}	8.39×10^{-7}	x
3	54	4.2×10^{-17}	5.2×10^{-10}	1.5×10^{-2}
5	150	5.6×10^{-17}	2.2×10^{-16}	5.3×10^{-4}
7	294	9.7×10^{-17}	6.9×10^{-18}	9.5×10^{-6}
9	486	8.3×10^{-17}	8.4×10^{-19}	5.9×10^{-8}
11	726	2.2×10^{-16}	3.2×10^{-18}	1.7×10^{-10}
13	1014	1.1×10^{-16}	1.2×10^{-17}	4.2×10^{-13}
15	1350	1.4×10^{-17}	4.8×10^{-18}	6.7×10^{-16}
17	1734	4.2×10^{-17}	8.9×10^{-18}	4.0×10^{-15}

* : Fonctions de Wachspress w_1 et w_5 à l'ordre 5

La même démarche est répétée pour les intégrales sur le bord. Une nouvelle fois, nous considérons des polynômes, un produit de sinus et un produit de fonctions de Wachspress. Il est important de remarquer que les fonctions de Wachspress dégénèrent en un polynôme sur le bord de l'hexagone. En particulier, les fonctions s'annulent sur les côtés opposés à leurs

nœuds d'ancrage. Dans ce cadre, une fonction de Wachspress w_i à l'ordre 5 va dégénérer en un polynôme d'ordre 5. Ainsi, la fonction w_i^2 est un polynôme d'ordre 10 sur l'arête où se situe son nœud d'ancrage. En pratique, il est compliqué de calculer analytiquement les intégrales w_i^2 , alors nous procédons de la manière suivante en étudiant une nouvelle fois un résidu :

$$i_{m,b}^i = \int_b w_i^2 ds = \sum_{l=1}^{N=6 \times m \times m} \omega_l w_i^2(\xi_l, \eta_l) \quad (3.14)$$

Bien que des quadratures d'ordre 6 auraient pu suffire, dans l'idée de comparer avec des bases polynomiales à des degrés élevés (potentiellement d'ordre 6-7), nous avons fait le choix d'utiliser des quadratures d'ordre 8 pour le bord. Le Tableau 3.3 illustre quelques résultats d'intégration sur le bord numéro 1 (voir Figure 1.2).

TABLEAU 3.3 Erreurs d'intégration numérique pour le bord numéro 1.

Ordre m	Nombre de points	$e_k = \int_b x^2 + y^2 - i_{x^2,y^2}^\dagger $	$e_k = \int_b \sin(x) \sin(y) - i_{\sin}^{\dagger'} $	$e_k^{2,2} = i_{k,b}^{2,2} - i_{k-1,b}^{2,2} $
2	54	2.8×10^{-17}	1.3×10^{-5}	1.7×10^{-4}
4	150	5.6×10^{-17}	1.2×10^{-11}	3.6×10^{-2}
6	294	5.6×10^{-17}	1.0×10^{-17}	1.9×10^{-3}
8	486	2.8×10^{-17}	2.1×10^{-17}	7.5×10^{-16}
10	726	2.8×10^{-17}	3.8×10^{-18}	4.4×10^{-16}

* : Fonctions w_2 à l'ordre 5

$$\dagger : i_{x^2,y^2} = \int_b x^2 + y^2 ds = \frac{p}{\sqrt{3}} \left(\frac{7h_s^2}{3} + \frac{h_s^2}{3} \right)$$

$$\dagger' : i_{\sin} = \int_b \sin(x) \sin(y) ds = \frac{h_s}{2} (\sin(\sqrt{3} - 2(h_p + h_s)) + \sin(\sqrt{3} + 2h_p + h_s) + \sin(\sqrt{3} + 2(h_s - h_p)) + \sin(\sqrt{3} + 2h_p - h_s))$$

Ainsi, les calculs suivants sont réalisés avec des quadratures d'ordre 15 pour les intégrales dans le volume et d'ordre 8 pour les intégrales sur le bord de l'hexagone.

2 Vérification des ordres de convergence et premières comparaisons avec la méthode des solutions manufacturées

Dans cette section, deux EDP sont résolues sur des maillages hexagonaux : l'équation de Poisson et une équation de transport de neutrons respectivement discrétisées par des méthodes d'éléments finis continus et discontinus. Pour s'assurer que les bases d'éléments finis sont adaptées aux schémas de discrétisation de ces équations, la méthode des solutions manufacturées [90] est utilisée pour caractériser les ordres de convergence à partir des estimateurs d'erreurs *a priori* [68, 89]. Des vérifications sont alors réalisées pour des solutions régulières en dehors de l'espace d'approximation. Suite à ces analyses, dans l'objectif de comparer les bases de polynômes et des bases de Wachspress pour la résolution de l'équation de transport, des études sont effectuées en considérant des solutions faiblement régulières appartenant au même espace que la solution de transport. Ces résultats fournissent des premiers éléments de comparaison avant de réaliser des calculs sur des cœurs de réacteur. Soulignons que des études similaires ont été réalisées dans [51] pour plusieurs bases d'éléments finis nodales d'ordre 1 et 2. Cependant, bien que les ordres de convergence soient retrouvés, il n'y avait pas de discussion autour des différences d'erreurs numériques produites par ces bases. Dans cette section, nous allons proposer une première analyse autour de ces différentes erreurs.

2.1 Principe général

En considérant le domaine ouvert \mathcal{D} de \mathbb{R}^2 avec pour frontière $\partial\mathcal{D}$, soit l'EDP suivante :

$$Lu(\vec{r}) = f(\vec{r}) \quad \forall \vec{r} \in \mathcal{D} \quad (3.15)$$

avec des conditions au bord de Dirichlet ou de Neumann, L un opérateur, f une source et u la solution associée de sorte que l'Equation (3.15) soit bien posée.

Les MMS consistent à fixer une solution analytique u_m et à la substituer dans le problème initial pour en déduire le membre de droite f et les conditions aux limites. Par conséquent, ce nouveau problème peut être résolu numériquement et conduit à une solution discrète u_h . La quantité d'intérêt est alors l'erreur entre u_h et u_m .

En pratique, un domaine rectangulaire \mathcal{D} ($L_x \times L_y$) maillé par des hexagones uniformes est considéré. Pour éviter d'effectuer des développements supplémentaires conséquents pour mailler le bord de \mathcal{D} avec des triangles, des quadrilatères ou des pentagones, le maillage est

restreint aux hexagones contenus dans \mathcal{D} , notons cet ensemble K . Ce choix est justifiable car ces développements ne sont pas requis pour l'application à des calculs de transport des neutrons dans un cœur.

Aussi, dans le cadre usuel de la méthode MMS, le domaine maillé ne change pas. Or ici, le domaine K est modifié à chaque raffinement en h ainsi que les conditions aux limites associées. Sans le prouver formellement, nous avons considéré que cette particularité n'aurait pas d'impact en régime asymptotique avec une norme globale telle que la norme L^2 . Nos résultats numériques ont par la suite confirmé cette hypothèse.

Par conséquent, nous considérons alors le problème modifié suivant à la place de l'Equation (3.15) :

$$\begin{cases} Lu(\vec{r}) = f(\vec{r}) & \forall \vec{r} \in K \\ u(\vec{r}) = u_m(\vec{r}) & \forall \vec{r} \in \partial K \end{cases} \quad (3.16)$$

où K désigne l'ensemble des éléments à l'intérieur de \mathcal{D} et ∂K le bord de ce maillage. Ce maillage est illustré sur la Figure 3.4. Dans ce problème modifié, des conditions aux limites de Dirichlet non homogènes sont utilisées avec u_m la solution exacte du problème initial donné par Equation (3.15).

Si l'on considère l'erreur en norme L^2 , $\varepsilon = \|u_m - u_h\|_{L^2(K)}$, celle-ci est bornée et devrait vérifier un résultat d'estimation d'erreur *a priori* sous la forme :

$$\|u_m - u_h\|_{L^2(K)} \leq Ch^c \quad (3.17)$$

où c est l'ordre de convergence attendu et C une constante positive indépendante de h . Dans les deux cas d'application rapportés par la suite, $c = \min(k+1, r)$ où k est l'ordre de la base, r la régularité de la fonction u_m (voir par exemple [89])¹. De manière équivalente, l'Equation (3.17) peut être réécrite en fonction du nombre de degrés de liberté N_{ddl} (c.f. par exemple, [51]) :

$$\|u_m - u_h\|_{L^2(K)} \leq C' N_{ddl}^{-\frac{\min(k+1, r)}{d}} \quad (3.18)$$

où d est la dimension de l'espace (ici $d = 2$) et C' une constante positive indépendante de N_{ddl} .

1. Ce résultat a été obtenu pour des triangles dans le cas de maillage régulier, il faudrait démontrer qu'il est valable dans le cas de maillages hexagonaux réguliers.

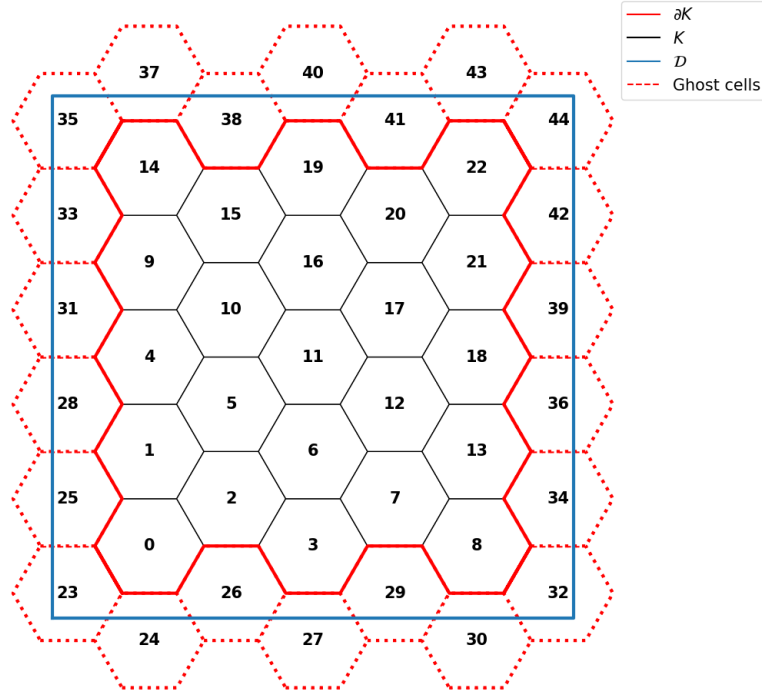


FIGURE 3.4 Exemple d'un domaine cartésien \mathcal{D} maillé avec des hexagones. Les lignes pointillées correspondent aux bords des cellules fantômes.

Ainsi, le calcul numérique de l'erreur $\|u_m - u_h\|_{L^2(K)}$ devrait conduire aux ordres de convergence théoriques en régime asymptotique.

Aussi, il est également intéressant d'analyser la convergence des erreurs d'approximation en régime pré-asymptotique en k au travers de l'estimation *a priori* :

$$\|u_m - u_h\|_{L^2(K)} \leq C'' k^{-\alpha} \quad (3.19)$$

où, comme il est expliqué dans [64], α est proche de r en régime pré-asymptotique et $\alpha = r - \frac{1}{2}$ en régime asymptotique.

2.2 Premières vérifications avec l'équation de Poisson

Les premières vérifications s'effectuent pour l'équation de Poisson discrétisée par des éléments finis continus. Cette équation sert de cas introductif aux fonctions de Wachspress avant de traiter une discrétisation plus complexe avec l'équation de transport. Rappelons que les bases de monômes orthogonalisées sont seulement applicables dans un cadre DG.

En effet, ces dernières ne permettent pas de garantir la continuité aux interfaces entre les différentes mailles. Par conséquent, elles ne seront pas considérées ici.

Dans cette section, les tests s'effectuent pour des fonctions dans l'espace d'approximation ainsi qu'avec une solution $u_m \in C^\infty(K)$, $u_m \notin P$.

Problème continu

Considérons l'équation de Poisson avec des conditions de Neumann homogènes :

$$\begin{cases} -\Delta u = f \text{ sur } \mathcal{D} \\ \frac{\partial u}{\partial \vec{n}} = 0 \text{ sur } \partial \mathcal{D} \end{cases} \quad (3.20)$$

En reprenant les notations précédentes, le problème modifié suivant sur le domaine K est considéré au lieu de l'Equation (3.20) :

$$\begin{cases} -\Delta u(\vec{r}) = f(\vec{r}) & \forall \vec{r} \in K \\ u(\vec{r}) = u_m(\vec{r}) & \forall \vec{r} \in \partial K \end{cases} \quad (3.21)$$

La démonstration de l'existence et l'unicité du problème issu de l'Equation (3.21) repose sur l'utilisation d'un relèvement $g \in H^2(K)$ tel que $g = u_m$ presque partout sur ∂K . En considérant la fonction $\tilde{u} = u - g$, cela conduit à $\tilde{u} = 0$ sur ∂K . Ceci permet d'écrire le problème suivant :

$$\begin{cases} -\Delta \tilde{u}(\vec{r}) = f(\vec{r}) + \Delta g(\vec{r}) & \forall \vec{r} \in K \\ \tilde{u}(\vec{r}) = 0 & \forall \vec{r} \in \partial K \end{cases} \quad (3.22)$$

En supposant $f \in L^2(K)$, la forme variationnelle peut être obtenue :

$$\int_K \vec{\nabla} \tilde{u}(\vec{r}) \vec{\nabla} v(\vec{r}) = \int_K f(\vec{r}) v(\vec{r}) + \int_K \Delta g(\vec{r}) v(\vec{r}) \quad \forall \vec{r} \in K, \quad \forall v \in H_0^1(K) \quad (3.23)$$

Il est alors possible de démontrer que \tilde{u} est l'unique solution l'Equation (3.23) et que les Équations (3.21) et (3.23) sont équivalentes [40]. Dans ce cadre, à partir de l'unicité de \tilde{u} , nous pouvons prouver que $u = \tilde{u} + g \in H^1(K)$ est l'unique solution de l'Equation (3.21) [40].

Problème discret

Pour obtenir une solution approchée, K est maillé en un ensemble de N_κ cellules hexagonales régulières κ_i , $K = \bigcup_{i=1}^{N_\kappa} \kappa_i$. L'espace d'approximation $V_h = \{v_i \in L^2(K), \forall \kappa_i \in K, v_{i|\kappa_i} \in P\}$ tel que $V_h \subset V$ avec P l'espace des fonctions de Wachspress. Dans ce cadre, la formulation faible approchée s'obtient en sommant les intégrales sur chaque maille κ_i et en prenant pour v une fonction $v_i \in P$ [40].

En notant le vecteur d'inconnues $\vec{x} = [u(a_i)]_{1 \leq i \leq N_{ddl}}$ où $a_i = (x_i, y_i)$ sont les nœuds du maillage, la matrice et le vecteur source peuvent alors s'écrire :

$$R = [R_{ij}]_{1 \leq i, j \leq N_{ddl}} \text{ avec } R_{ij} = \begin{cases} \sum_{i=1}^{N_\kappa} (\vec{\nabla} w_i, \vec{\nabla} w_j)_{L_2(\kappa_i)} & \text{si } a_i, a_j \notin \partial K \\ 1 & \text{si } i = j \text{ avec } a_i, a_j \in \partial K \\ 0 & \text{si } i \neq j \text{ avec } a_i \text{ ou } a_j \in \partial K \end{cases} \quad (3.24)$$

$$\forall i \in \llbracket 1, N_{ddl} \rrbracket, b_i = \begin{cases} (f, w_i)_{L_2(\kappa_i)} - \sum_{a_j \in \partial K} R_{ji} u_m(a_j) & \text{si } a_i \notin \partial K \\ u_m(a_i) & \text{si } a_i \in \partial K \end{cases}$$

$$\vec{b} = [b_i]_{1 \leq i \leq N_{ddl}}^T \quad (3.25)$$

En combinant les Equations (3.24) et (3.25), le système linéaire suivant est créé :

$$R\vec{x} = \vec{b} \quad (3.26)$$

Premières vérifications avec $u_m \in P_k$

En considérant une solution manufacturée $u_m \in P_k$, si $P_k \subset P$, alors théoriquement :

$$\|u_m - u_h\|_{L^2(K)} = 0$$

Le Tableau 3.4 illustre les erreurs numériques obtenues pour différentes fonctions u_m aux ordres 1 à 5. Pour toutes les bases rationnelles d'ordre égal ou supérieur à k , on a

$e_h = \|u_m - u_h\|_{L^2(K)}$, $\sup(e_h) \leq 10^{-10}$ ce qui est en accord avec la théorie. Également, les erreurs augmentent lorsque l'ordre augmente.

TABLEAU 3.4 Quelques résultats numériques sur le calcul de e_h pour différents u_m avec $N_K = 9$.

Ordre	N_{ddl}	$u_m = 1 + x + y$	$u_m = x^3 + 4.27xy - 0.25y^2$	$u_m = x^4 + 0.6y^5 - 1.98x^3y^2 + 0.75y^3x$
1	54	1.23×10^{-15}	6.54×10^{-1}	3.01×10^{-1}
2	108	1.40×10^{-14}	3.64×10^{-2}	3.64×10^{-2}
3	162	1.56×10^{-13}	2.17×10^{-13}	1.21×10^{-1}
4	216	1.98×10^{-12}	5.87×10^{-12}	9.09×10^{-3}
5	270	1.51×10^{-11}	3.16×10^{-11}	6.21×10^{-11}

Vérification à l'aide d'une fonction $u_m \in C^\infty(K)$, $u_m \notin P$

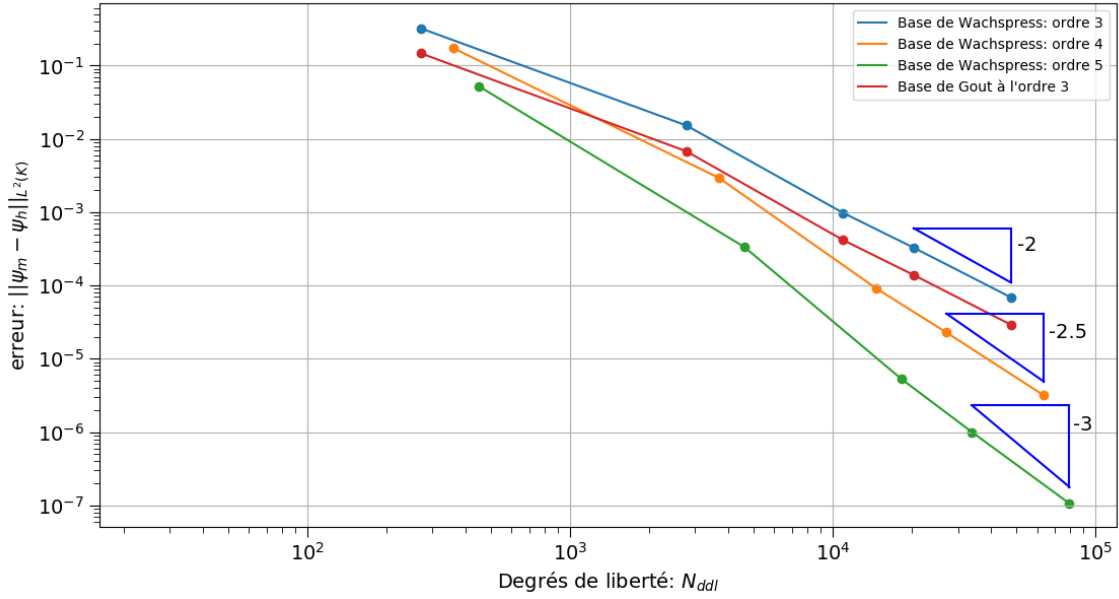
Afin de vérifier les propriétés des bases de Wachspress construites par la méthode développée dans la section 3 du chapitre deux, nous avons considéré pour u_m la fonction régulière suivante :

$$u_m = \sin\left(\frac{2\pi x}{L_x}\right) \sin\left(\frac{2\pi y}{L_y}\right) \cos\left(\frac{2\pi x}{L_x}\right) \cos\left(\frac{2\pi y}{L_y}\right) + 1$$

De telle sorte que, quel que soit l'ordre k de la base, les ordres de convergence asymptotiques par rapport à h (resp. N_{ddl}) donnés par l'Equation (3.17) (resp. l'Equation (3.18)) sont toujours toujours égaux à $k + 1$ (resp. $-\frac{k+1}{d}$). Par conséquent, les courbes d'erreur et la comparaison entre les ordres de convergence obtenus avec les ordres théoriques permettent de vérifier que les bases obtenues respectent l'estimateur *a priori* de l'Equation (3.17).

Analysons maintenant les résultats numériques pour $\mathcal{D} = [-L_x/2, L_x/2] \times [-L_y/2, L_y/2]$ avec $L_x = 20$, $L_y = 30$ et pour des pas $h \in \{10, 5, 2, 1, 0.5\}$. Sur la figure 3.5, l'erreur ε est représentée en fonction de N_{ddl} en considérant les solutions obtenues à l'ordre k allant de 3 à 5. À l'ordre 3, en plus de la base calculée par notre méthode (en fixant les variables indépendantes à 0), la base proposée par Gout a également été utilisée. On peut observer que, dans tous les cas, les ordres de convergence théoriques (représentés par des triangles sur la figure) sont approximativement retrouvés, démontrant alors la validité des bases proposées.

Un point important sur l'impact du choix des variables indépendantes est visible ici en analysant les résultats de notre base à l'ordre 3 et celle proposée par Gout. En effet, bien que les ordres de convergence soient les mêmes, les erreurs d'approximation sont différentes. Cela provient de la constante C issue de l'Equation (3.18) qui est inférieure avec la base de

FIGURE 3.5 Courbes de l'erreur e_h pour différents ordres

Gout (environ 3 fois) à la nôtre. Ce résultat peut se justifier par les propriétés que présente la base de Gout *i.e.* des symétries géométriques similaires à celles de l'hexagone. *A contrario*, lors de la construction de la base où les variables libres ont été arbitrairement fixées à zéro, aucune contrainte n'a été imposée pour respecter ces propriétés. Étant donné que la fonction u_m présente des symétries par rapport à x et y , les symétries de la base de Gout s'avèrent utiles pour réduire l'erreur d'approximation. Ainsi, comme expliqué dans les perspectives du chapitre 2, la méthode de construction de base proposée pourrait être modifiée de sorte que les contraintes de symétrie soient ajoutées à l'ensemble des équations.

2.3 Une deuxième vérification et des premières comparaisons pour la résolution de l'équation de transport

Considérons l'équation S_n -monogroupe pour une direction $\vec{\Omega}_n$ sur le domaine K .

$$\begin{cases} \vec{\Omega}_n \cdot \vec{\nabla} \psi_n(\vec{r}) + \Sigma_t(\vec{r}) \psi_n(\vec{r}) = Q_n(\vec{r}) & \forall \vec{r} \in K \\ \psi_n(\vec{r}) = \psi_n^{BC}(\vec{r}) & \forall \vec{r} \in \partial K \end{cases} \quad (3.27)$$

L'enjeu ici est double : une première étape de vérification où, de manière similaire à l'équation de Poisson, nous nous assurons que la résolution de l'Equation (3.27) avec

les bases de Wachspress et les bases de monômes orthogonalisées conduit aux ordres de convergence attendus pour des solutions régulières. La deuxième étape consiste à considérer des solutions dont les régularités varient et sont caractéristiques de l'équation de transport. En particulier, les solutions faiblement régulières choisies appartiennent au même espace que la solution de l'équation de transport *i.e.* $H^{\frac{1}{2}-\varepsilon}$ ou $H^{\frac{3}{2}-\varepsilon}$ (en fonction des conditions aux limites). Pour ces cas là, l'ordre de la base n'a pas d'impact sur l'ordre de convergence.

Il s'agit alors d'étudier uniquement les erreurs numériques entre les deux bases d'éléments finis pour l'hexagone. Nous nommerons dans toute la suite **HOW** (High Order Wachspress) les bases de Wachspress et **POLY** les bases de monômes orthogonalisées.

Vérifications pour des solutions dans l'espace d'approximation

De même que pour l'équation de Poisson, des fonctions polynomiales dans l'espace d'approximation sont considérées pour les bases des fonctions de Wachspress et les bases de monômes orthogonalisées.

Les Tableaux 3.5 et 3.6 illustrent les erreurs numériques $e_h = \|\psi_m - \psi_h\|_{L^2(K)}$ respectivement obtenues pour POLY et HOW.

Les résultats sont similaires à ceux sur l'équation de Poisson *i.e.* $e_h = \|\psi_m - \psi_h\|_{L^2(K)}$, $\sup(e_h) \leq 10^{-10}$ pour les deux bases utilisées. Également, les erreurs d'approximation augmentent lorsque l'ordre augmente pour HOW contrairement à POLY. Comme expliqué précédemment, les bases de Wachspress calculées par notre méthode sont bornées par la précision pour laquelle les contraintes géométriques et les projections sont vérifiées, d'où des erreurs autour de 10^{-10} à l'ordre 5.

On notera toutefois que HOW présentent des erreurs d'approximation inférieures à POLY pour les ordres 1 à 3. Aussi, lorsque l'ordre de ψ_m est supérieur à l'ordre de la base, les erreurs d'approximation sont inférieures avec HOW qu'avec POLY.

TABLEAU 3.5 Erreurs e_h pour différents ψ_m avec les bases de monômes orthogonalisées

Ordre	N_{unk}	Erreurs e_h avec POLY pour différentes fonctions ψ_m		
		$\psi_m = 1 + x + y$	$\psi_m = x^3 + 4.27xy - 0.25y^2$	$\psi_m = x^4 + 0.6y^5 - 1.98x^3y^2 + 0.75y^3x$
1	27	2.72×10^{-13}	2.66×10^1	1.56×10^2
2	54	2.53×10^{-13}	2.28×10^0	2.86×10^1
3	90	2.71×10^{-13}	6.41×10^{-13}	3.50×10^0
4	135	2.71×10^{-13}	7.18×10^{-13}	1.81×10^{-1}
5	189	2.86×10^{-13}	6.33×10^{-13}	1.92×10^{-12}

TABLEAU 3.6 Erreurs e_h pour différents ψ_m avec les bases de Wachspress

Ordre	N_{ddl}	Erreurs e_h avec HOW pour différentes fonctions ψ_m		
		$\psi_m = 1 + x + y$	$\psi_m = x^3 + 4.27xy - 0.25y^2$	$\psi_m = x^4 + 0.6y^5 - 1.98x^3y^2 + 0.75y^3x$
1	54	4.32×10^{-15}	6.71×10^{-1}	3.61×10^0
2	108	1.50×10^{-14}	2.01×10^{-2}	2.24×10^{-1}
3	162	7.51×10^{-14}	2.65×10^{-13}	5.90×10^{-1}
4	216	5.76×10^{-12}	1.34×10^{-11}	1.07×10^{-2}
5	270	1.39×10^{-11}	3.35×10^{-11}	5.86×10^{-11}

Vérification des ordres de convergence $\psi_m \in C^\infty(\mathcal{D}), \psi_m \notin P$

Considérons maintenant une solution $\psi_m \in C^{+\infty}(\mathcal{D}), \psi_m \notin P$. À cette fin, la fonction suivante a été choisie :

$$\psi_m(x, y) = \sin\left(\frac{2\pi x}{L_x}\right) \sin\left(\frac{2\pi y}{L_y}\right) \quad (3.28)$$

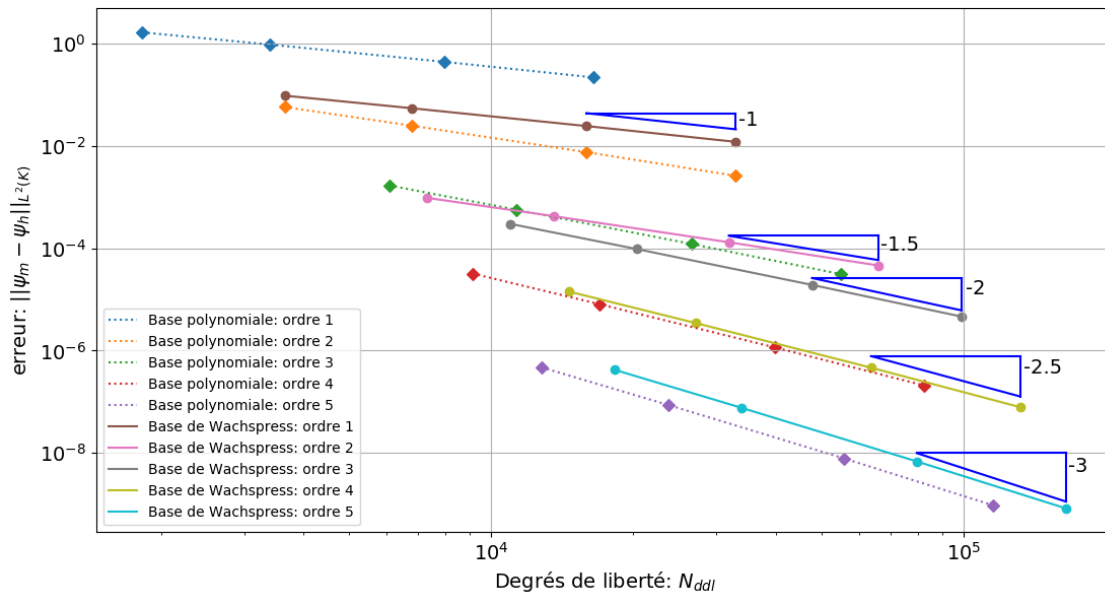


FIGURE 3.6 $e_h = \|\psi_m - \psi_h\|_{L^2(K)}$ sur le domaine K aux ordres 1-2-3-4-5 pour les bases de monômes orthogonalisées et de Wachspress. Les droites présentant des \diamond et des \circ correspondent respectivement aux résultats pour POLY et HOW.

La Figure 3.6 représente l'erreur e_h en fonction des N_{ddl} pour chaque ordre k de 1 à 5 avec les deux différentes bases. Celles-ci donnent bien les ordres de convergence attendus. Bien que le résultat obtenu avec la base de Gout à l'ordre 3 ne soit pas représenté sur la figure, le constat réalisé en comparant cette base et la nôtre à l'ordre 3 pour l'équation de Poisson est le même ici. En effet, la base de Gout conduit une nouvelle fois à des erreurs

moins grandes (approximativement deux fois inférieures). Les symétries vérifiées par la base de Gout permettent de réduire les erreurs d'approximation pour cette fonction ψ_m .

Concernant la comparaison en elle-même entre POLY et HOW, les résultats sont similaires à ceux des fonctions ψ_m dans l'espace d'approximation. Typiquement, les différences sont significatives jusqu'à l'ordre 3. Entre autres, à l'ordre 3, pour le même nombre de mailles, l'écart entre POLY et HOW est presque d'un ordre de grandeur. Également, toujours pour les ordres 1 à 3, si une comparaison est effectuée pour un nombre de degrés de liberté équivalent, HOW génère des erreurs inférieures à POLY.

Ce constat n'est plus valable pour les ordres 4 et 5. En effet, à partir de l'ordre 4, les différences sont bien moins visibles. Si pour le même nombre de mailles à l'ordre 4, HOW produit des erreurs trois fois inférieures à POLY, pour le même nombre de degrés de liberté, les erreurs sont quasiment identiques. Dans le cas de l'ordre 5, les erreurs sont presque égales pour le même nombre de mailles. Cependant, pour le même nombre de degrés de liberté, les erreurs avec POLY sont inférieures à celles de HOW.

Cela peut s'expliquer par la constante C dans l'estimateur *a priori* qui dépend en réalité de l'ordre k de la base. L'évolution de l'erreur (et donc de C) selon k est différente entre les deux bases. Dans le cas de POLY, cette erreur est initialement plus grande que celle de HOW mais décroît plus vite en comparaison de HOW lorsque k augmente. Ce résultat est mis en évidence si une régression linéaire est effectuée sur les erreurs e_h à un pas h donné pour chaque ordre $k \in \llbracket 1, 5 \rrbracket$ pour POLY et HOW comme l'illustre la Figure 3.7. Pour ce cas où $\psi_m \in C^\infty(\mathcal{D})$, que ce soit pour POLY ou HOW, la convergence en k est, comme attendue, très rapide. Au-delà de ce constat général, il est intéressant de noter que cette erreur est plus importante pour k faible avec POLY qu'avec HOW mais décroît aussi plus vite de telle sorte à devenir plus faible pour k plus grand.

Enfin, bien que cela ne soit pas visible ici, la précision de la base de Wachspress est bornée à l'ordre 5 dans le sens où l'erreur d'approximation ne diminue plus et se limite autour de 10^{-10} . Ce phénomène a pu être observé lors des cas précédents en considérant des fonctions dans l'espace d'approximation. En revanche, ce résultat n'est pas visible pour la base de monômes dont les erreurs d'approximation peuvent être inférieures à 10^{-12} . Ce phénomène serait certainement visible pour POLY en augmentant le nombre de degrés de liberté de sorte à obtenir des erreurs d'approximation autour de 10^{-12} . En effet, pour cette plage d'erreur, POLY est aussi limitée par la précision pour laquelle elle vérifie les erreurs de projection. Par conséquent, les erreurs ne diminueraient plus.

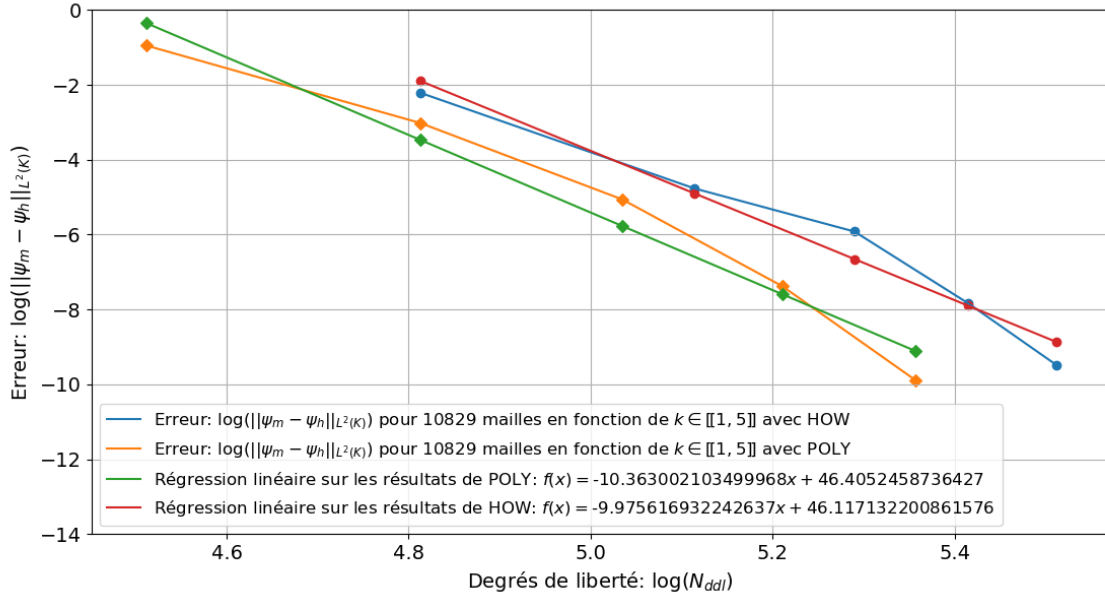


FIGURE 3.7 Variation de l'erreur $\|\psi_m - \psi_h\|_{L^2(K)}$ selon k pour un nombre de cellules $N_K = 10829$ pour la fonction $\psi_m(x, y) = \sin\left(\frac{2\pi x}{L_x}\right) \sin\left(\frac{2\pi y}{L_y}\right)$ avec POLY (\diamond) et HOW (\circ).

Premières comparaisons dans le cas de solutions faiblement régulières

Dans l'optique de comparer les erreurs numériques entre les bases de Wachspress et les bases de monômes orthogonalisées pour des calculs de neutronique sur des réacteurs, des premières études sont réalisées via des solutions dont la régularité est similaire à la régularité globale des solutions de l'équation de transport. À cette fin, les fonctions présentées pour la première fois dans le travail de thèse de Duo [33] sont utilisées. Ces fonctions sont solutions de l'équation de transport pour une direction de propagation $\vec{\Omega}$ avec une source nulle. L'Equation (3.29) décrit cette fonction analytique selon notre domaine :

$$\psi_m(x, y, \vec{\Omega}_1) = \begin{cases} \psi_l e^{-\Sigma_t(x + \frac{L_x}{2})/a}, & (y + L_y/2) > (x + L_x/2) \frac{b}{a} \\ \psi_b e^{-\Sigma_t(y + \frac{L_y}{2})/b}, & (y + L_y/2) < (x + L_x/2) \frac{b}{a} \end{cases} \quad (3.29)$$

où $\Omega_1 = [a, b]^T$ avec $a = \cos(\pi/3)$, $b = \sin(\pi/3)$ et $\psi_l, \psi_b \in \mathbb{R}$.

La Figure 3.8 représente les conditions aux limites imposées sur le domaine ainsi que la ligne caractéristique issue de l'équation $y = (x + L_x/2) \frac{b}{a} - L_y/2$.

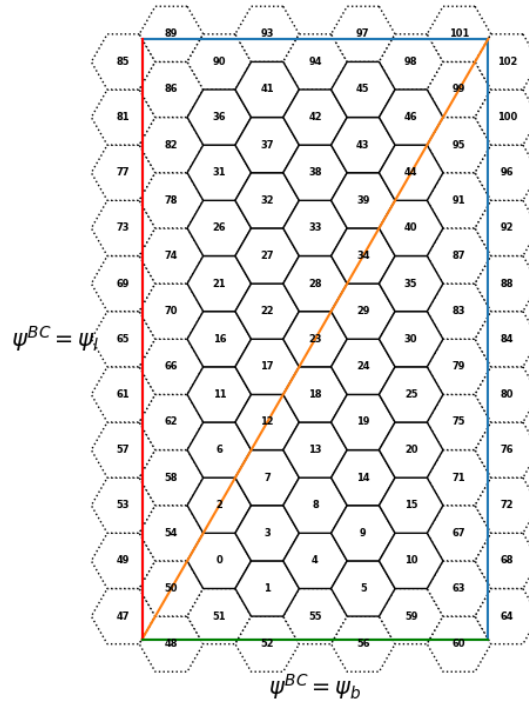


FIGURE 3.8 Conditions aux limites imposées sur les bords gauche et inférieur du domaine. La ligne droite orange correspond à la ligne caractéristique d'équation $y = (x + L_x/2)\frac{b}{a} - L_y/2$.

De plus, les conditions aux limites peuvent être discontinues selon les valeurs de ψ_l et ψ_b , ainsi deux cas sont distingués et nommés MMS1 et MMS2.

- Le premier cas, appelé MMS1, correspond à $\psi_l \neq \psi_b$ où $\psi_l = 1$ et $\psi_b = 0$, la solution appartient à $H^{\frac{1}{2}-\varepsilon}$ (voir [60, 71]). La solution ψ_m est discontinue le long de la ligne caractéristique.
- Le second cas, nommé MMS2, correspond à $\psi_l = \psi_b = 1$, la solution appartient à $H^{\frac{3}{2}-\varepsilon}$ (également selon [60, 71]). Cette fois-ci, la solution ψ_m est continue mais sa dérivée est discontinue perpendiculairement à la ligne caractéristique.

Comme expliqué en début de section, l'intégration sur l'hexagone s'effectue via un partitionnement en six triangles (la Figure 3.9c illustre cette idée), où des quadratures de Gauss-Legendre en $x - y$ sont appliquées pour chaque triangle. Bien que ces quadratures soient adaptées pour intégrer des fonctions polynomiales (voir section 1 du chapitre 3), les discontinuités des solutions MMS1-MMS2 impactent l'intégration et donc le calcul d'erreur en norme L^2 . Ce phénomène a pu être observé lors nos premiers calculs en considérant un domaine où les points de quadrature pouvaient se situer sur la ligne caractéristique. Ce domaine est appelé domaine non aligné et est représenté sur la Figure 3.9a. Dans ce cas, la convergence était impossible en dépit du nombre élevé de degrés de liberté.

L'enjeu de ce travail étant l'analyse de convergence, nous ne cherchons pas à adapter notre quadrature pour traiter les discontinuités (voir [77]). Ainsi, pour remédier à ce problème, une idée a consisté à aligner le domaine le long de la ligne caractéristique comme le montre la Figure 3.9b. Par conséquent, ces dernières sont situées le long des arêtes des triangles partitionnant chaque hexagone, les points d'intégration ne sont alors plus localisés sur la discontinuité. Dans ce cadre, nous pouvons nous focaliser sur l'étude qui nous intéresse. Notons que le domaine aligné garantit bien une régularité globale de $H^{\frac{1}{2}-\varepsilon}$ pour MMS1 et de $H^{\frac{3}{2}-\varepsilon}$ pour MMS2 car des mailles hexagonales sont traversées par la ligne caractéristique².

Précisons également qu'une vérification de l'intégration de la fonction MMS1 a été effectuée en étudiant la variation d'un résidu r_m en fonction de nombre de points m de quadrature. En introduisant $i_m^{MMS1} = \int_K \psi_m(x, y) dV$ et ψ_m la fonction MMS1, il a été observé que le résidu $r_m = |i_m^{MMS1} - i_{m-1}^{MMS1}| \leq 10^{-15}$ pour une quadrature d'ordre neuf³. Par conséquent, avec le maillage aligné, l'erreur d'intégration devient, comme dans les cas précédents, négligeable de telle sorte que l'étude de convergence en raffinement de maillage peut être menée correctement.

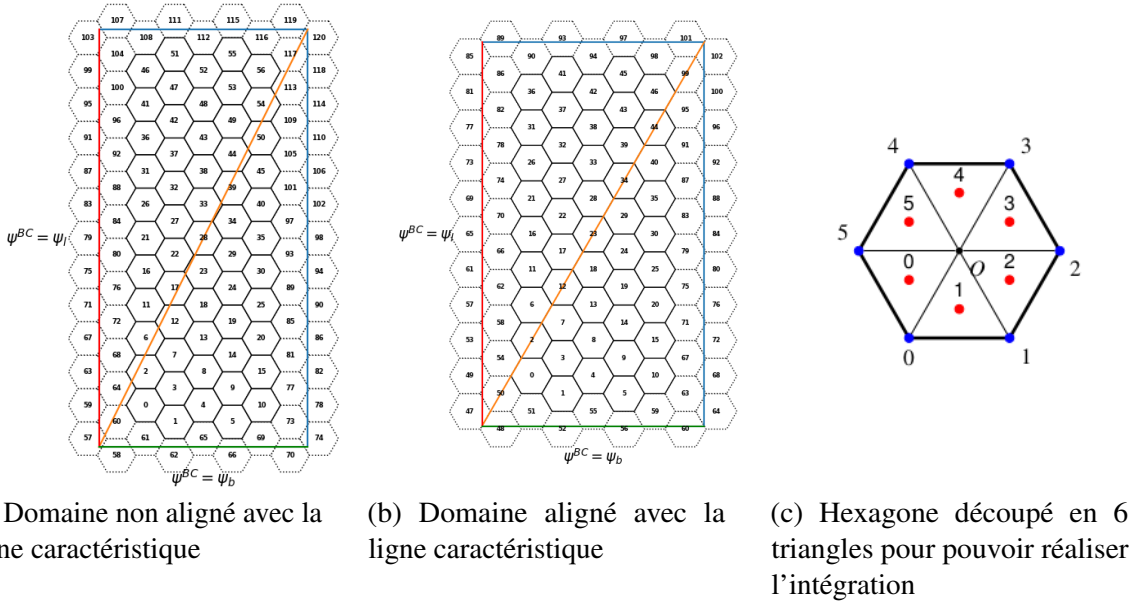


FIGURE 3.9 Illustrations des domaines non aligné et aligné ainsi que du partitionnement d'un hexagone en six triangles.

2. Dans le cas d'un maillage triangulaire aligné le long de la ligne caractéristique, la convergence n'est alors plus limitée par la régularité globale de la solution exacte [51]

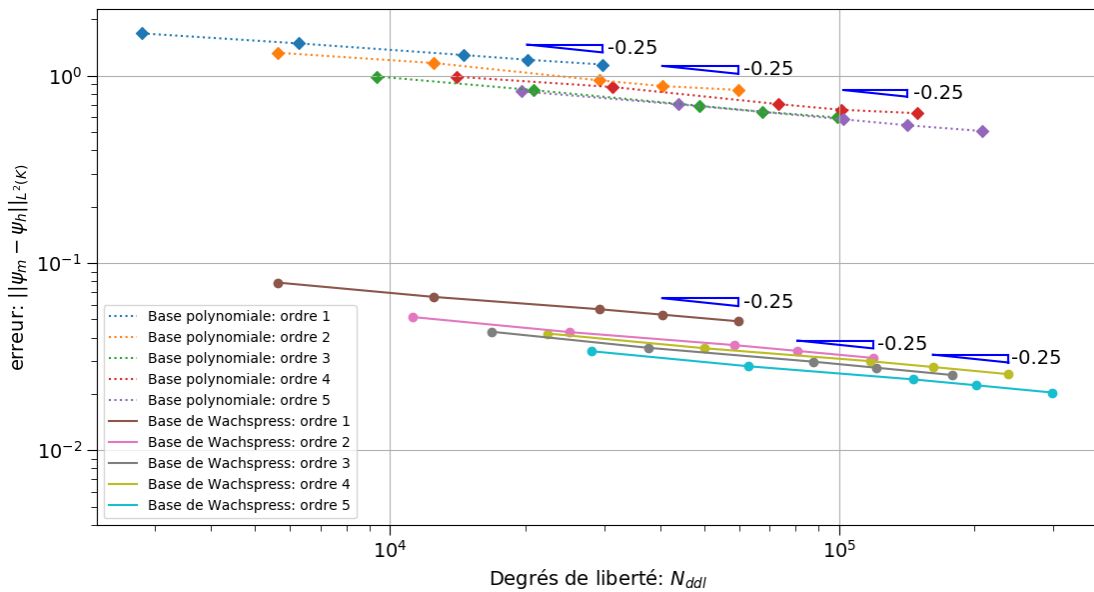
3. Dans le cas du domaine non aligné, le résidu r_m ne converge pas malgré l'augmentation de l'ordre de la quadrature.

Comme dans les cas précédents, les erreurs e_h sont représentées sur les Figures 3.10a et 3.10b respectivement pour les fonctions MMS1 et MMS2 en fonction des N_{ddl} pour les solutions obtenues aux ordres k de 1 à 5 avec les deux bases.

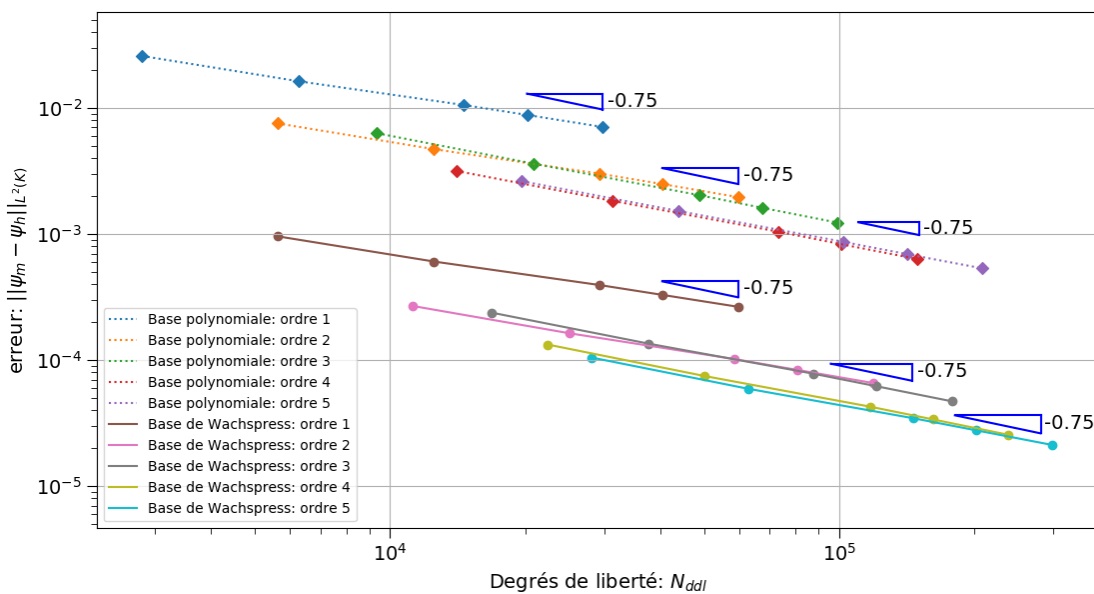
De manière générale, les ordres de convergence théoriques sont approximativement retrouvés pour HOW et POLY comme l'illustrent les triangles des Figures 3.10a et 3.10b. Par conséquent, notre schéma de discrétisation avec ces deux bases d'éléments finis permet de converger vers la solution de l'équation de transport en régime asymptotique.

Aussi, pour analyser la convergence en k dans le régime pré-asymptotique⁴, les Figures 3.11 et 3.12 représentant les erreurs à e_h pour un h fixé pour différents ordres k avec HOW et POLY respectivement pour les fonctions MMS1 et MMS2 sont tracées. Pour rappel, l'estimateur *a priori* de l'Equation (3.19) conduit à un ordre de convergence α proche de r en régime pré-asymptotique et $\alpha = r - \frac{1}{2}$ en régime asymptotique. Dans notre cas, on observe que, comme attendu, la constante α obtenue est alors proche de la régularité des solutions MMS1 et MMS2 pour les deux bases obtenues.

4. Le régime asymptotique en k serait atteint pour des ordres supplémentaires.



(a) Erreur $\|\psi_m - \psi_h\|_{L^2(K)}$ pour $k \in \llbracket 1, 5 \rrbracket$ pour la fonction MMS1 avec POLY et HOW



(b) Erreur $\|\psi_m - \psi_h\|_{L^2(K)}$ pour $k \in \llbracket 1, 5 \rrbracket$ pour la fonction MMS2 avec POLY et HOW

FIGURE 3.10 Illustrations de l'erreur $\|\psi_m - \psi_h\|_{L^2(K)}$ pour les fonction MMS1-MMS2 avec POLY (droites présentant des \diamond) et de Wachspress (droites présentant des \circ).

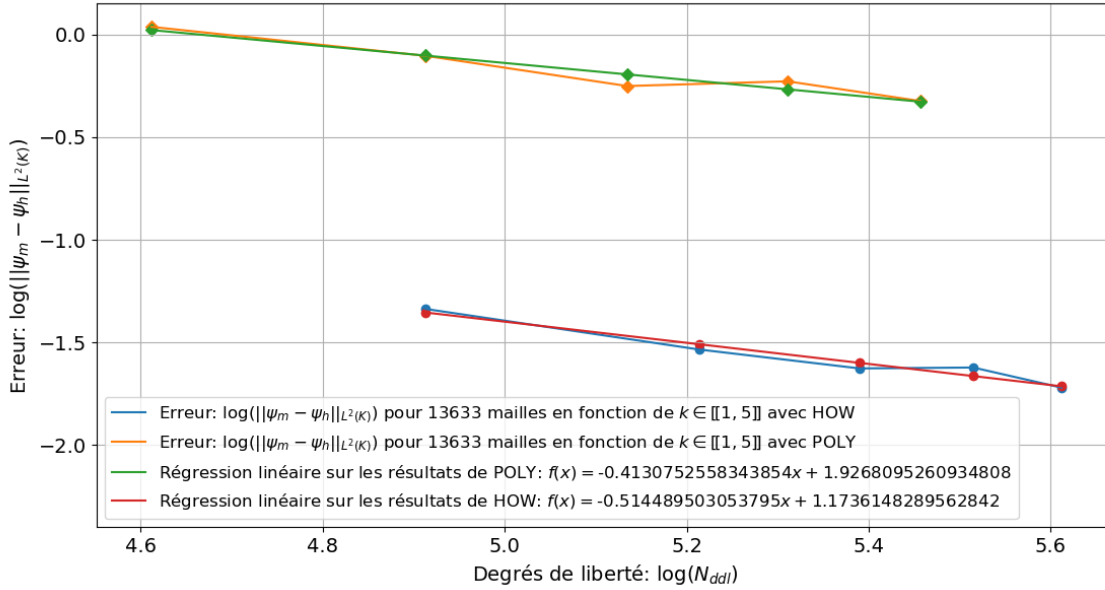


FIGURE 3.11 Variation de l'erreur $\|\psi_m - \psi_h\|_{L^2(K)}$ selon k pour un nombre de cellules $N_K = 13633$ pour la fonction MMS1 avec POLY (\diamond) et HOW (\circ).

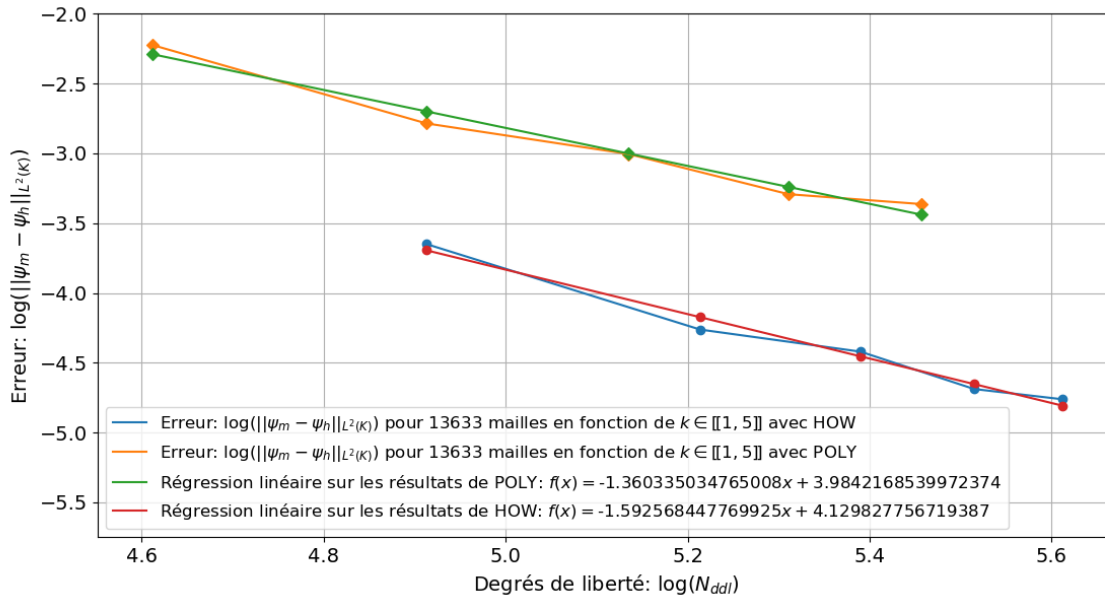


FIGURE 3.12 Variation de l'erreur $\|\psi_m - \psi_h\|_{L^2(K)}$ selon k pour un nombre de cellules $N_K = 13633$ pour la fonction MMS2 avec POLY (\diamond) et HOW (\circ).

Concernant la comparaison sur les erreurs numériques, contrairement au cas régulier, les écarts ici sont très prononcés entre HOW et POLY.

Pour le même nombre de mailles, pour un ordre k donné, il y a environ un ordre de grandeur

d'écart entre HOW et POLY que ce soit pour MMS1 ou MMS2. Une nouvelle fois, ce point est en lien avec la constante C intervenant dans l'estimateur *a priori*. Ainsi, pour ces fonctions faiblement régulières, les degrés de liberté supplémentaires offerts par les fonctions de Wachspress permettent d'obtenir de meilleures approximations. Une remarque peut être ajoutée ici, puisque la fonction ψ_m ne possède pas de propriétés de symétrie (que ce soit pour MMS1 ou MMS2), la base de Gout produit des erreurs sensiblement égales à la nôtre à l'ordre 3.

Aussi, la régularité de la fonction ψ_m a un impact sur les erreurs d'approximation. En effet, que ce soit pour POLY ou HOW, les erreurs obtenues avec MMS2 sont inférieures de deux à trois ordres de grandeur⁵ à celles de la fonction MMS1. Ce résultat s'explique une nouvelle fois par la constante C intervenant dans l'Equation (3.17). Il y aurait donc un lien entre la régularité de solution et la constante C .

Également, les erreurs numériques pour les bases de Wachspress et de polynômes sont plus prononcées pour la fonction MMS1 que la fonction MMS2. Or, la fonction MMS2 est plus régulière (pour rappel, cette fonction est continue sur tout le domaine \mathcal{D}) que la fonction MMS1. Dans ce cadre, l'espace généré par les fonctions de Wachspress s'avère utile pour réduire les erreurs d'approximation dans le cas de fonctions faiblement régulières. Plus la fonction est régulière, moins les différences sont marquées.

Afin d'aller plus loin dans l'interprétation de ces résultats numériques, deux perspectives paraissent intéressantes. D'abord, il faudrait étudier plus en détail le comportement de la constante C en fonction de la régularité de la fonction ψ_m . Dans ce cadre, des études supplémentaires pourraient être réalisées via les MMS en considérant des fonctions avec différentes régularités, comme des fonctions $C^1(K)$, $C^2(K)$. Elles permettraient de quantifier numériquement C pour différentes fonctions ψ_m et pourraient servir de point de départ pour initier une démarche théorique vis-à-vis de cette dépendance de C à r . Ensuite, il serait intéressant d'analyser les variations de C (ou C') en fonction de la base choisie. Dans ce cadre, une étude théorique sur les erreurs d'interpolation pourrait être réalisée. Gout a fourni différents résultats sur ces erreurs pour les fonctions de Wachspress [4, 5, 46]. Cela constituerait alors un point de départ. Une démarche relativement similaire pourrait être envisagée pour les bases de monômes orthogonalisées pour l'hexagone régulier. Cependant, les propriétés de ces deux bases diffèrent. En effet, la base de monômes est une base modale là où les bases de Wachspress sont des bases de Lagrange. Par conséquent, les résultats obtenus par Gout ne peuvent pas être exactement appliqués sur des bases de monômes. Ainsi, des études bibliographiques plus conséquentes seraient nécessaires. Une première piste possible serait d'étudier la constante de Lebesgue [27]. En effet, l'erreur d'interpolation polynomiale

5. En fonction de l'ordre k de la base.

dépend de cette constante qui elle-même dépend à son tour des points d'interpolation. Il serait alors intéressant de trouver un moyen de quantifier cette constante pour cette base de monômes orthogonalisée.

Après ces vérifications et ces comparaisons de nos bases sur ces différents cas MMS, nous allons maintenant effectuer des comparaisons pertinentes pour l'application visée au travers de calculs de cœurs (en termes de k_{eff} et de taux d'absorption) issus de benchmarks de la littérature.

3 Résultats numériques sur deux benchmarks numériques

Suite aux vérifications précédentes, l'équation de transport de neutrons est maintenant résolue sur deux benchmarks de réacteurs à géométrie hexagonale.

Le premier correspond à un benchmark monogroupe ([16]) dont les résultats ont déjà conduit à une publication [61]. Le second benchmark correspond au quatrième modèle décrit par Takeda dans [100] comportant quatre groupes. Pour l'étudier en deux dimensions, nous nous sommes placés à une côte axiale correspondant à une hauteur de 95 cm. Dans ce plan, les barres de contrôle sont totalement insérées.

Pour ces deux benchmarks, des comparaisons sont faites en termes de k_{eff} et de taux d'absorption. À cette fin, les solutions sont calculées entre notre solveur et des solveurs issus du CEA. L'objectif étant de se focaliser sur la discrétisation spatiale, les résultats sont tous obtenus pour les mêmes quadratures angulaires⁶. Aussi, pour l'ensemble des solveurs, la variable spatiale est discrétisée par une méthode de Galerkin discontinue avec un schéma décentré amont. Les différences proviennent des maillages et des bases utilisés. Pour le benchmark monogroupe et le benchmark multigroupe, une solution de référence et des solutions plus grossières sont respectivement obtenues par le solveur APOLLO3[®]/MINARET [91] et par le solveur SNATCH [43]. Précisons que pour les benchmarks issus de [16] et de [100], les critères d'arrêt pour les itérations internes, multigroupes et de la puissance sont de 10^{-6} .

Soulignons que les géométries définies dans ces benchmarks ne sont pas convexes, en conséquence, des neutrons peuvent sortir et revenir à l'intérieur du domaine, ce qui est, en soi,

6. Les quadratures angulaires considérées sont des quadratures produits (cf. section 1.4 du chapitre 1) aux ordres (2,3) et (3,4) pour les benchmarks issus de [16] et de [100]. La notation (2,3) signifie que l'on effectue une quadrature de Gauss-Legendre d'ordre 2 pour l'angle polaire et une quadrature de Gauss-Chebyshev d'ordre 3 pour l'angle azimutal.

contraire aux conditions aux limites. Pour remédier à cela, il faudrait compléter le bord avec des mailles triangulaires par exemple ; on notera, qu'en pratique, lors des calculs de cœurs de réacteurs à assemblages hexagonaux, cette correction est rarement appliquée. Notons aussi que ce problème associé à la non-convexité n'était pas présent lors de nos études avec les MMS de par nos choix vis-à-vis de la direction angulaire choisie $\vec{\Omega}_n$ et de l'alignement du domaine maillé par rapport à cette direction.

3.1 Application à un benchmark monogroupe

Description

Ce benchmark consiste en un cœur composé de trois différents matériaux associés à différentes valeurs de sections efficaces comme illustré sur la Figure 3.13 et sur le Tableau 3.7⁷. Comme l'explique A. Hébert [54], ce benchmark n'est pas représentatif d'un problème réel mais a été conçu pour amplifier les effets de transport et d'anisotropie dans le domaine. Pour ce cœur hexagonal, la longueur d'une arête d'une cellule est de 19 cm⁸.

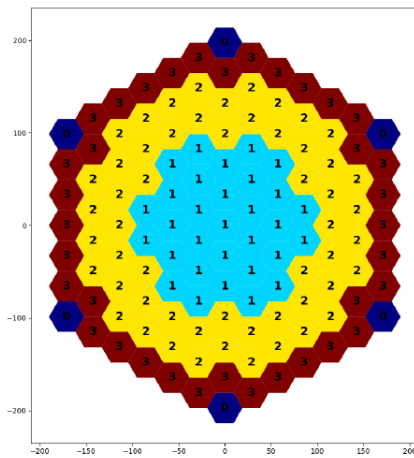


TABLEAU 3.7 Sections efficaces

Reactions [cm ⁻¹]	1	2	3
Σ_t	0.025	0.025	0.075
Σ_{s0}	0.013	0.024	0.0
Σ_{s1}	0.0	0.006	0.0
$\nu\Sigma_f$	0.0155	0.0	0.0

FIGURE 3.13 Benchmark hexagonal 2D

Résultats numériques

De manière similaire à [16], ce benchmark monogroupe est analysé dans deux contextes différents : à l'aide du solveur APOLLO3[®]/MINARET et à l'aide de bases d'éléments

7. Le matériau numéroté zéro est ici du vide pour compléter la description du cœur hexagonal.

8. Ce qui correspond à un pas du réseau hexagonal de $19\sqrt{3}$.

finis développées pour l'hexagone via notre solveur maquette. En utilisant MINARET, une solution de référence est calculée par un sous-maillage d'une cellule hexagonale en cinquante triangles. Cette solution génère un facteur de multiplication effectif de référence k_{eff} de 1.002135 ainsi que des taux d'absorption de référence. Également, pour se comparer à des solutions plus grossières de MINARET, d'autres solutions sont obtenues en maillant un hexagone en six triangles⁹ pour les ordres 1 à 5 - cette solution est nommée MS1. Aussi, les calculs sont réalisés pour les ordres 1 à 5 avec les bases des fonctions de Wachspress (HOW pour rappel), auxquels s'ajoute la base de Gout à l'ordre 3, et les bases de monômes orthogonalisées (POLY). Chaque calcul est caractérisé par le nombre total d'inconnues N_{ddl} associé à la discrétisation spatiale, et correspond à $N_K \times N(k)$, N_K étant le nombre de cellules du maillage et $N(k)$ le nombre de fonctions par cellule.

D'après le Tableau 3.8, lorsque l'ordre augmente, pour MS1, HOW et POLY, les valeurs de k_{eff} convergent vers la valeur de référence. Si l'on compare POLY et HOW à MS1, les différences sont faibles à partir de l'ordre 4. En particulier, le Δk_{eff} vaut 0.1 pcm avec MS1 à l'ordre 5, alors que HOW et POLY induisent respectivement des Δk_{eff} de 0.45 pcm et de -0.6 pcm avec quatre et sept fois moins d'inconnues. Cependant, bien que les erreurs soient plus prononcées entre MS1 et HOW/POLY pour les ordres inférieurs à 4, il est important de noter que MS1 contient un grand nombre de degrés de liberté en comparaison de HOW ou POLY.

Ainsi, si une comparaison est faite pour un nombre de degrés de liberté équivalent, les solutions HOW et POLY produisent des Δk_{eff} inférieurs à ceux de MS1. Typiquement, la variable N_{ddl} à l'ordre 5 de POLY est comparable à celle de MS1 à l'ordre 1. Or $\Delta k_{eff} = 9.7$ avec MS1 à l'ordre 1 et $\Delta k_{eff} = -0.6$ à l'ordre 5 avec POLY. Un constat similaire peut être effectué en comparant l'ordre 3 de HOW avec l'ordre 1 de MS1.

Concernant la comparaison entre POLY et HOW, ces deux bases génèrent approximativement les mêmes résultats. Les écarts sont visibles à l'ordre 1 et à l'ordre 3. En particulier pour l'ordre 1, POLY produit un Δk_{eff} de -78.1 contre un Δk_{eff} de 46.9 avec HOW. En revanche, avec moins de degrés de liberté, POLY produit des erreurs plus petites que HOW. Par exemple, la variable N_{ddl} est la même à l'ordre 1 pour HOW et à l'ordre 2 pour POLY et on observe que POLY induit un Δk_{eff} de -2.6 alors que HOW génère un Δk_{eff} de 46.9.

Pour compléter cette analyse, le k_{eff} étant une grandeur globale, nous étudions alors des variables locales à l'aide des taux d'absorption par assemblage. Le Tableau 3.9 montre les différents résultats obtenus.

9. Cela correspond au sous-maillage minimal d'une cellule hexagonale en triangles par MINARET.

TABLEAU 3.8 Comparaison du Δk_{eff}

Ordre	MS1		HOW		POLY	
	$N_{ddl} = N_K \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_K \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_K \times N(k)$	Δk_{eff} [pcm]
1	2286 = 762 × 3	9.7	762 = 127 × 6	46.9	381 = 127 × 3	-78.1
2	4572 = 762 × 6	-0.5	1524 = 127 × 12	-2.7	762 = 127 × 6	-2.6
3*	7620 = 762 × 10	0.5	2286 = 127 × 18	-2.57	1270 = 127 × 10	6.2
4	11430 = 762 × 15	0.1	3048 = 127 × 24	-0.6	1905 = 127 × 15	0.3
5	16002 = 762 × 21	0.1	3810 = 127 × 30	0.4	2667 = 127 × 21	-0.6

* $\Delta k_{eff} = -1.8$ pour la base de Gout

TABLEAU 3.9 Comparaison des erreurs sur les taux d'absorption

Ordre	N_{ddl}	MS1		N_{ddl}	HOW		N_{ddl}	POLY	
		max [-]	rms [-]		max [-]	rms [-]		max [-]	rms [-]
1	2286	9.7×10^{-3}	1.9×10^{-2}	762	5.1×10^{-2}	1.9×10^{-2}	381	5.4×10^{-2}	2.1×10^{-2}
2	4572	7.5×10^{-3}	1.6×10^{-3}	1524	1.1×10^{-3}	7.3×10^{-4}	762	1.7×10^{-3}	3.3×10^{-3}
3*	7620	6.7×10^{-3}	1.4×10^{-3}	2286	5.4×10^{-3}	2.1×10^{-3}	1270	5.6×10^{-3}	2.2×10^{-3}
4	11430	4.0×10^{-4}	2.7×10^{-4}	3048	1.5×10^{-3}	6.9×10^{-4}	1905	1.5×10^{-3}	1.0×10^{-3}
5	16002	6.1×10^{-4}	1.5×10^{-4}	3810	8.5×10^{-4}	6.0×10^{-4}	2667	1.1×10^{-3}	3.7×10^{-4}

* Les erreurs max et rms valent respectivement 1.3×10^{-3} et 1.1×10^{-3} pour la base de Gout.

De la même manière que pour le k_{eff} , lorsque l'ordre augmente, les erreurs sur les taux d'absorption convergent vers la valeur de référence.

Par ailleurs, dans le cas de l'erreur maximale, en dehors de l'ordre 4, les ordres de grandeur sont les mêmes avec HOW, POLY et MS1. Pour l'ordre 4, il y a une différence de 7×10^{-4} entre HOW, POLY et MS1.

Pour l'erreur rms, MS1, POLY et HOW convergent vers les valeurs de référence lorsque l'ordre augmente. En particulier, les ordres de grandeur sont les mêmes entre MS1 et HOW.

Aussi, notons que les erreurs rms générées par HOW sont inférieures à celles obtenues par POLY pour les ordres 2 et 4. La différence la plus flagrante étant à l'ordre 2 avec un écart d'à peu près 3.6×10^{-3} . Pour le cas de l'ordre 5, POLY produit une erreur rms presque deux fois inférieure à celle de HOW. Cependant, de manière générale, les erreurs rms induites par POLY et HOW sont relativement similaires. Une explication provenant de la régularité du flux peut justifier ce résultat. Pour rappel, La régularité globale de la solution exacte est, dans le cas de ce benchmark, de $\frac{3}{2} - \epsilon$. Toutefois, dans la comparaison entre les bases, les écarts importants observés dans le cas de MMS2 ne sont pas retrouvés malgré une solution présentant la même régularité. Cette différence est à attribuer à la régularité locale qui n'est pas uniforme. Plus précisément, dans les cas réacteurs (selon la disposition des matériaux et les sections efficaces mises en jeu), les zones où la solution est significativement affectée

par la perte de régularité associée aux discontinuités entre matériaux ¹⁰ peuvent être limitées de telle sorte que la convergence avec les différentes bases s'apparente plus à ce qui a été observé avec la solution manufacturée régulière. Ce constat vis-à-vis de la régularité locale est étayé par les résultats rapportés dans la littérature ([41] où une estimation de la régularité locale est faite (dans une logique de raffinement hp afin de choisir entre h et p).

Aussi, bien que les calculs entre MS1 et HOW/POLY ne puissent pas être directement comparés en termes de temps de calcul à ce stade, la comparaison de N_{ddl} donne une indication du gain potentiel que l'on peut attendre avec des éléments finis développés pour l'hexagone. Ceci est illustré plus en détails à la Fig. 3.14 où les écarts par assemblage sont présentés pour la solution MS1 à l'ordre 1 et la solution HOW à l'ordre 3, toutes deux ont le même nombre de variables spatiales $N_{ddl} = 2286$.

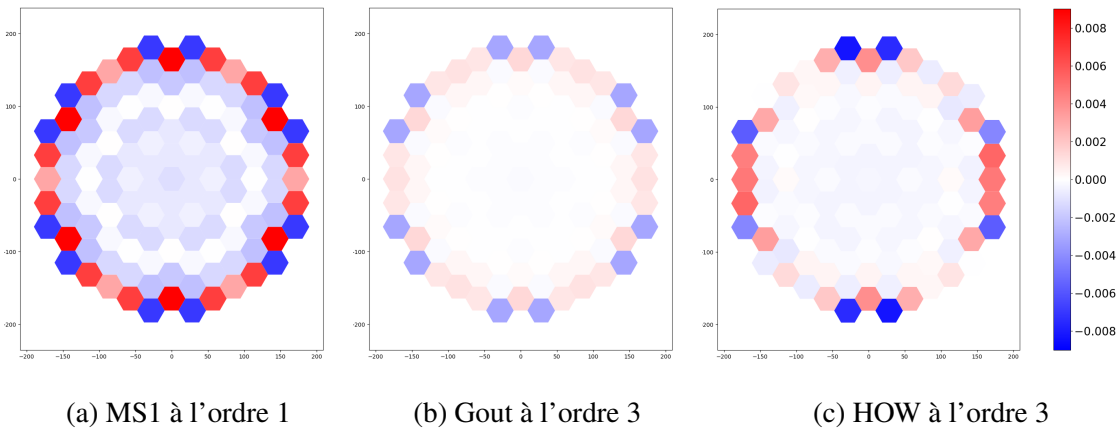


FIGURE 3.14 Erreurs sur les taux d'absorption pour $N_{ddl} = 2286$

Une remarque importante peut être faite en comparant les solutions de HOW et de Gout à l'ordre 3. Bien que les bases d'éléments rationnels associées vérifient les mêmes propriétés de l'Equation (2.6), les résultats rapportés dans les Tableaux 3.8 et 3.9 diffèrent significativement. Cela provient de la différence des espaces générés par les deux bases, même si ces dernières engendrent P_3 . En d'autres termes, comme expliqué auparavant, le choix des variables indépendantes pour générer des bases de Wachspress à un ordre k donné change l'espace d'approximation et a un impact notable sur l'erreur d'approximation.

Un dernier commentaire peut être ajouté sur la Fig. 3.14. En effet, les erreurs produites par HOW présentent une légère asymétrie. Ce constat n'est pas apparent aux ordres 1 et 2 où aucun degré de liberté n'est disponible lors du choix des fonctions rationnelles. Cela provient probablement du choix arbitraire de fixer les variables indépendantes à 0.

10. Le long des lignes caractéristiques de direction $\vec{\Omega}_n$ issues des discontinuités géométriques.

3.2 Application à un benchmark à quatre groupes d'énergie

Description du benchmark

Considérons maintenant le quatrième modèle décrit par Takeda dans [100]. Ce benchmark est constitué de huit matériaux comme illustré sur la Figure 3.15 et les données concernant les matrices de diffusion et les sections efficaces sont disponibles dans [100]. Pour rappel, nous nous sommes placés à la cote de hauteur 95 cm avec les barres de contrôles insérées. Précisons que pour ce modèle, la longueur d'une arête d'un assemblage hexagonal est de 7.50 cm.

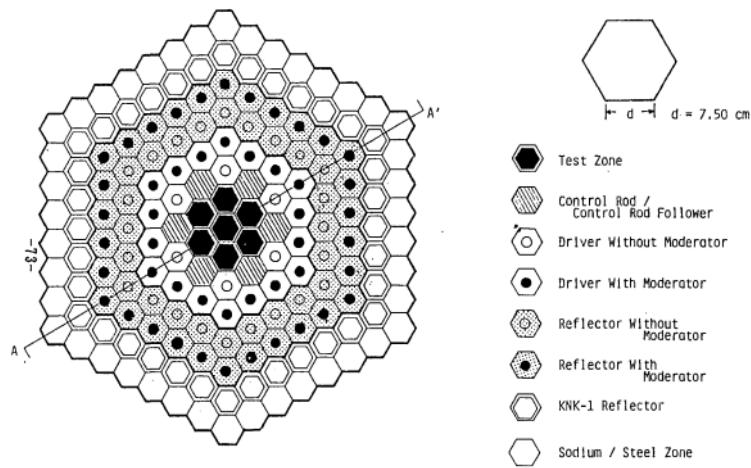


FIGURE 3.15 Benchmark hexagonal 2D issu de [100]

Résultats numériques

Comme dans la section 3.1 précédente, l'analyse est réalisée par deux solveurs - à l'aide du solveur SNATCH et de notre solveur maquette. Premièrement une solution de référence est obtenue avec SNATCH, celle-ci est générée à partir d'une subdivision d'un hexagone en $3 \times 6 \times 6$ losanges et en considérant des fonctions de base d'ordre 5. Cette solution produit un k_{eff} de référence de 1.009815 et des taux d'absorption de référence. Également, des solutions plus grossières de SNATCH sont calculées pour les ordres 1 à 5 en divisant un hexagone en 3 losanges¹¹ - cette solution est nommée SNS1. Pareillement à la section précédente, les calculs sont effectués avec les bases de Wachspress (HOW) (auxquelles s'ajoutent une nouvelle fois la base de Gout à l'ordre 3) et avec les bases de polynômes orthogonalisées

11. Cela correspond au maillage minimal d'un hexagone par SNATCH.

(POLY). Nous considérons les mêmes notations qu'auparavant, c'est-à-dire que les calculs sont caractérisés par le nombre de degrés de liberté N_{ddl} associé à la discrétisation spatiale où $N_{ddl} = N_{\kappa} \times N(k)$, avec N_{κ} le nombre de cellules et $N(k)$ le nombre de fonctions de base par maille. La comparaison sur les taux d'absorption se fera sur ces grandeurs multigroupes condensées à un groupe d'énergie.

Étudions dans un premier temps les différences sur le k_{eff} , les résultats sont disponibles dans le Tableau 3.10. Lorsque l'ordre augmente, les différentes solutions SNS1, HOW, POLY convergent vers le k_{eff} de référence. On remarquera toutefois que les erreurs sont plus prononcées ici que pour le benchmark précédent. Aussi, à partir de l'ordre 3, POLY produit des solutions plus proches de la solution de référence que HOW. C'est notamment visible à l'ordre 3 où POLY induit un Δk_{eff} de 9.8 contre 32.5 avec HOW. Aussi, il est intéressant de souligner que pour un nombre d'inconnues similaire, POLY et HOW produisent des Δk_{eff} inférieurs à ceux de SNS1. En particulier, les différences sont flagrantes entre l'ordre 3 de POLY et l'ordre 1 de SNS1 où les Δk_{eff} respectifs sont de 9.8 pcm et de 779.5 pcm. Un constat équivalent peut être effectué entre l'ordre 2 de HOW et l'ordre 1 de SNS1. Également, on observe qu'à l'ordre 5 POLY génère un Δk_{eff} de -0.1 qui est inférieur à celui de SNS1 alors que cette solution contient environ 5 fois plus d'inconnues. Pour compléter l'analyse, nous étudions une nouvelle fois les erreurs sur les taux d'absorption.

TABLEAU 3.10 Comparaison du Δk_{eff}

ordre	SNS1		HOW		POLY	
	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]
1	2028 = 507 × 4	779.5	1024 = 169 × 6	1721.2	507 = 169 × 3	-2277
2	4563 = 507 × 9	13.0	2028 = 169 × 12	-16.6	1024 = 169 × 6	199.6
3	8112 = 507 × 16	4.7	3042 = 169 × 18	32.5	1690 = 169 × 10	9.8
4	12675 = 507 × 25	-0.4	4056 = 169 × 24	-11.1	2535 = 169 × 15	-2.3
5	18252 = 507 × 36	-0.5	5070 = 169 × 30	5.1	3549 = 169 × 21	-0.1

* $\Delta k_{eff} = 26.6$ pour la base de Gout

Pour toutes les bases SNS1, HOW, POLY, les erreurs sur les taux d'absorption convergent vers les valeurs de référence. Cependant, à partir de l'ordre 3, les erreurs produites par SNS1 sont inférieures d'un ordre de grandeur à celles de POLY et de HOW. Il est toutefois intéressant de comparer les erreurs rms obtenues à l'ordre 5 avec POLY et HOW et à l'ordre 3 avec SNS1. Bien que POLY et HOW aient un N_{ddl} inférieur à SNS1, les erreurs rms sont très similaires. En effet, les écarts entre SNS1-POLY et SNS1-HOW sont respectivement d'un facteur 1.5 et de 2 pour cette erreur. Aussi, concernant les différences entre POLY et HOW, contrairement au cas du k_{eff} , il est visible ici que HOW produit des erreurs rms inférieures

TABLEAU 3.11 Comparaison des erreurs sur les taux d'absorption

ordre	N_{ddl}	SNS1		N_{ddl}	HOW		N_{ddl}	POLY	
		max [-]	rms [-]		max [-]	rms [-]		max [-]	rms [-]
1	2028	3.0×10^{-2}	1.4×10^{-2}	1024	6.2×10^{-2}	3.1×10^{-2}	507	1.5×10^{-1}	7.5×10^{-2}
2	4563	6.6×10^{-4}	5.7×10^{-4}	2028	8.8×10^{-4}	9.7×10^{-4}	1024	1.1×10^{-2}	5.5×10^{-3}
3*	8112	2×10^{-4}	1.6×10^{-4}	3042	3.8×10^{-3}	1.4×10^{-3}	1690	2.3×10^{-3}	3.4×10^{-3}
4	12675	4.3×10^{-5}	2.4×10^{-5}	4056	9.1×10^{-4}	3.3×10^{-4}	2535	7.6×10^{-4}	6.5×10^{-4}
5	18252	9.1×10^{-5}	3.5×10^{-5}	5070	1.6×10^{-3}	3.2×10^{-4}	3549	8.8×10^{-4}	2.4×10^{-4}

* Les erreurs max et rms valent respectivement 4.57×10^{-3} et 1.86×10^{-3} pour la base de Gout

à POLY jusqu'à l'ordre 4. Notons en particulier le cas de l'ordre 2 où l'erreur rms est de 5.5×10^{-3} avec POLY contre 9.7×10^{-4} avec HOW. En moyenne, les erreurs rms obtenues avec HOW sont deux fois inférieures à celles de POLY. Ce résultat n'est plus vrai à partir de l'ordre 5 où POLY produit une solution légèrement plus proche de la solution de référence que celle de HOW (à un facteur 1.36 près). Également, dans le cas de l'erreur max, à partir de l'ordre 3, celle-ci est plus élevée avec HOW qu'avec POLY, en particulier à l'ordre 5. Ces résultats peuvent se justifier une nouvelle fois par la régularité du flux. Le constat réalisé pour le benchmark précédent sur ce point est toujours valable ici.

Une analyse supplémentaire sur la convergence en k semble être intéressante ici. En effet, les erreurs max et rms se détériorent pour SNS1 de l'ordre 4 à l'ordre 5. Cela n'est pas visible sur le benchmark précédent. Ces résultats pourraient indiquer que l'on sort du régime de convergence pré-asymptotique en k plus tôt dans ce benchmark. En conséquence, le fait de raffiner uniquement en k avec HOW et POLY devient moins favorable que dans le cas du benchmark précédent.

Enfin, notons ici que les erreurs se situent majoritairement sur le bord du réacteur, plus particulièrement sur ses coins comme l'illustre la Figure 3.16. Ce constat est également observable avec SNS1 (bien que moins marqué). En particulier, les matériaux KNK1 reflector et Sodium/Steel Zone (les deux dernières couronnes) correspondent aux deux matériaux où sont majoritairement situées les erreurs.

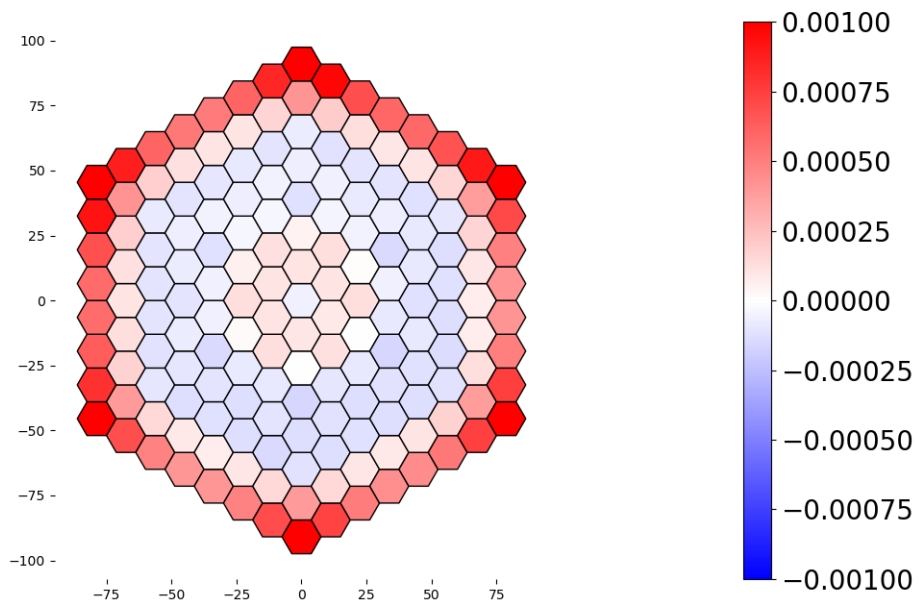


FIGURE 3.16 Erreur sur les taux d'absorption avec HOW à l'ordre 5

4 Conclusion et perspectives

Dans ce chapitre, nous avons vérifié que les bases développées dans le chapitre précédent sont adaptées à la résolution d'EDP par une méthode d'éléments finis. Cette vérification a été réalisée par la méthode des solutions manufacturées en considérant des solutions $\psi_m \in C^\infty(K)$, $\psi_m \notin P$. Pour ces fonctions, les différentes bases produisent les ordres de convergence attendus.

Suite à ces vérifications, des premières comparaisons entre HOW et POLY ont été réalisées par les MMS via des fonctions correspondant à des solutions de l'équation de transport pour une direction $\vec{\Omega}$ donnée avec une source nulle [33]. Les ordres de convergence théoriques ont été obtenus et plus particulièrement, les bases HOW produisent des erreurs bien inférieures¹² pour tous les ordres à POLY. En particulier, moins la solution est régulière, plus les bases de Wachspress présentent une erreur moindre que les bases de monômes orthogonalisées.

Pour approfondir les comparaisons et répondre à l'objectif initial de cette thèse, deux benchmarks de cœurs ont été calculés à l'aide de nos bases et comparés (en termes de k_{eff} et de taux d'absorption) à des solutions de référence produites par les solveurs APOLLO3[®]/MINARET et SNATCH. Les bases de Wachspress et les bases de monômes convergent vers les k_{eff} et les taux de référence lorsque l'ordre augmente. Un résultat fort porte sur la comparaison des erreurs en fonction du nombre de degrés de liberté. En effet, à un nombre de degrés équivalent, les erreurs d'approximation produites par les bases développées sur l'hexagone sont inférieures à celles obtenues avec des triangles ou des lozanges. Ainsi, l'objectif de départ est atteint *i.e.* développer une discrétisation capable de traiter l'hexagone sans sous maillage et ainsi améliorer le compromis entre précision et coût de calcul.

En perspective, des analyses plus approfondies pourraient être effectuées sur la constante C intervenant dans l'estimateur *a priori*. En particulier, il serait intéressant de caractériser les variations de cette constante en fonction de la régularité de la solution à approcher. Dans ce cadre, des tests MMS pourraient être effectués en considérant d'autres solutions (typiquement des solutions dans $C^1(K)$, $C^2(K)$).

De même, une étude pourrait être réalisée pour quantifier la constante C en fonction de la base d'éléments finis choisie en étudiant l'erreur d'interpolation. Une bibliographie plus approfondie serait nécessaire. Nous pouvons toutefois citer des résultats connus [4, 46] pour les erreurs d'interpolation avec les bases de Wachspress. Pour le cas de la base de monômes orthogonalisée, une piste serait d'étudier la constante de Lebesgue [27] car l'erreur d'interpolation dépend de cette constante qui elle-même dépend des points d'interpolation.

12. D'au moins un ordre de grandeur.

Chapitre 4

Extension des bases hexagonales 2D à la 3D

De nouvelles bases 3D sont développées à partir des bases 2D obtenues précédemment. Dans ce cadre, un maillage cylindrique droit à base hexagonale est considéré de telle sorte qu'il est possible de combiner des fonctions de base radiales 2D et des fonctions de base 1D (le long de l'axe d'extrusion). Dans un premier temps, nous étudierons la base générée par le produit tensoriel entre ces deux ensembles de fonctions pour générer un espace polynomial tridimensionnel d'ordre k (P_k^3)¹. Ce choix est couramment utilisé dans le cadre de maillage prismatique en neutronique [9, 107]. Cependant, une remarque peut être ajoutée : la base 3D résultante génère un espace plus grand *i.e.* l'espace des polynômes dont les monômes sont de degrés partiels inférieurs ou égaux à k (Q_k^3). Afin de réduire la taille de la base considérée pour ces calculs 3D, une alternative a été proposée dans le solveur MINARET/APOLLO3®[63]. Cette approche heuristique (dénommée “option additive” dans MINARET) consiste à sommer P_k^2 et P_l^1 pour former l'espace d'approximation P , avec les ordres k et l choisis par l'utilisateur. De manière générale, bien que l'espace P_k^3 ne soit pas inclus dans cet espace P , cette option est privilégiée en pratique car elle offre un meilleur compromis entre la précision et le temps de calcul que le produit tensoriel (aussi implémenté dans MINARET sous le nom “option multiplicative”) [2]. Partant de ce constat, une composition originale de bases 2D et 1D est proposée dans ce chapitre afin de construire une base d'éléments finis 3D qui, tout en permettant d'engendrer l'espace P_k^3 , minimise le nombre de degrés de liberté nécessaires. Ces travaux ont notamment pu être présentés lors de la *27th International Conference on Transport Theory - ICTT27*.

1. Notons P_k^d l'espace des polynômes d'ordre k en dimension d . Cette définition est importante car les espaces générés par la suite dépendent de la dimension.

Notations

Considérons un domaine spatial prismatique droit (selon z) dont le maillage est construit à partir d'un maillage polygonal dans le plan $x - y$ extrudé selon différents plans dans z . Chaque maille κ de ce domaine est un prisme droit de base polygonale κ_{2D} dans le plan $x - y$ qui s'étend sur $\kappa_{1D} = [z_1, z_2]$ selon z . Notons $W_k = \{w_i^k(x, y)\}_{i \in \llbracket 1, N(k) \rrbracket}$ et $V_l = \{\phi_j(z)\}_{j \in \llbracket 0, l \rrbracket}$ respectivement les espaces des fonctions de base d'ordre k et d'ordre l pour κ_{2D} et κ_{1D} . En particulier, on a : $P_k^2 \subset \text{Vect}(W_k)$ et $P_l^1 \subset \text{Vect}(V_l)$.

1 Bases d'éléments finis 3D à l'aide du produit tensoriel

Pour obtenir une base de fonctions pour κ , un choix intuitif est le produit tensoriel des deux bases suivantes [39, 45] :

$$P = \text{Vect}(W_k \otimes V_l) = \text{Vect}(\{w_i(x, y)\phi_j(z)\}_{(i, j) \in \llbracket 1, N(k) \rrbracket \times \llbracket 0, l \rrbracket}) \quad (4.1)$$

Cette base est nommée **base multiplicative** et le solveur associé **solveur multiplicatif**. Il est intéressant de noter que cette base permet de générer des espaces dont les ordres sont mixtes *i.e.* l'ordre de la base 2D peut être différent de la base 1D. Également cet espace forme une base de $P_k^2 \otimes P_l^1$ car $P_k^2 \subset \text{Vect}(W_k)$ et $P_l^1 \subset \text{Vect}(V_l)$ et P constitue une famille libre.

Dans ce cadre, on peut définir l'élément fini (κ, P, Σ) suivant :

- κ : une maille 3D correspondant à un hexagone extrudé.
- P : espace d'approximation contenant $N_{k,l} = \text{card}(W_k) \times \text{card}(V_l)$ fonctions de base définies sur κ . Pour simplifier l'écriture, notons les fonctions de base v_i ,
 $P = \text{Vect}(\{v_i\}_{i \in \llbracket 1, N_{k,l} \rrbracket})$
- Σ : ensemble de formes linéaires sur P .

Pour une fonction f donnée, l'espace Σ s'écrit :

$$\Sigma = \{\forall i \in \llbracket 0, N_{k,l} \rrbracket, \sigma_i(f) = f_i\} \quad (4.2)$$

et les f_i se déduisent à partir de la résolution d'un système linéaire obtenu par projection de f dans P :

$$[\sigma_i(f)]_{1 \leq i \leq N_{k,l}} = \underline{\underline{M_\kappa}}^{-1} \left[\langle f, v_q \rangle_{L^2(\kappa)} \right]_{1 \leq q \leq N_{k,l}} \quad (4.3)$$

où $\underline{M_K}$ est la matrice de masse associée à la base $\{v_i\}_{i \in \llbracket 1, N_{k,l} \rrbracket}$

On peut montrer que Σ est P -unisolvent. Premièrement, $\dim(\Sigma) = \dim(P)$ et il ne reste plus qu'à vérifier que pour toutes les fonctions $\sigma_i \in \Sigma$ et $v_j \in P$ avec $(i, j) \in \llbracket 1, N_{k,l} \rrbracket^2$, $\sigma_i(v_j) = \delta_{i,j}$. Ce résultat est facilement obtenu puisque

$$[\sigma_i(f)]_{i,j \in \llbracket 1, N_{k,l} \rrbracket^2} = \underline{M_K}^{-1} \left[\langle v_j, v_q \rangle_{L^2(\kappa)} \right]_{j,q \in \llbracket 1, N_{k,l} \rrbracket^2} = \underline{M_K}^{-1} \underline{M_K} \quad (4.4)$$

Quelques remarques peuvent être ajoutées sur le nombre de degré de liberté $N_{k,l}$ selon si W_k est généré à partir des fonctions de Wachspress ou des polynômes.

Si W_k est constitué des fonctions de Wachspress :

$$N_{k,l} = \text{card}(W_k) \times \text{card}(V_l) = 6k \times (l+1) \quad (4.5)$$

Si W_k est constitué par des polynômes :

$$N_{k,l} = \text{card}(W_k) \times \text{card}(V_l) = \frac{(k+1)(k+2)}{2} \times (l+1) \quad (4.6)$$

Il est intéressant de noter que pour $k = l < 9$ (resp. $k = l \geq 9$), le nombre de fonctions de base de P sera plus grand (resp. petit) si W_k est constitué des fonctions de Wachspress plutôt que des fonctions polynomiales.

D'un point de vue pratique, l'espace V_l et le calcul des différentes matrices nécessaires à la résolution de l'équation de transport sont détaillés en Annexes 1 et 2. Il est important de souligner que le choix de l'espace V_l est arbitraire, une base de Lagrange ou de Legendre aurait également pu être utilisée.

Comme expliqué en début de chapitre, pour $k = l$, si l'on souhaite générer l'espace P_k^3 à partir de cette base multiplicative, l'espace obtenu est en fait Q_k^3 . Par conséquent, dans l'idée d'engendrer P_k^3 avec le moins de degrés de liberté possible, une nouvelle base 3D a été développée.

2 Une composition originale de bases d'éléments finis

Dans cette section, nous allons définir notre nouvel espace d'approximation P . Premièrement, en utilisant le caractère hiérarchique de P_l^1 , P_k^3 est décomposée en une somme de produits tensoriels des espaces P_m^2 et P_l^1 . Cette équation permettra alors dans un second temps de montrer que P_k^3 est inclus dans la somme de produits tensoriels des espaces $\text{Vect}(W_m)$ et $\text{Vect}(V_{k-m})$ qui sera notre espace P . Introduisons maintenant les bases hiérarchiques car celles-ci sont la notion clé de cette partie.

2.1 Une base polynomiale hiérarchique

Une famille $\{B_k\}_{k \geq 0}$, où B_k est un ensemble de polynômes, est dite hiérarchique si, $\forall k \geq 0$:

- B_k est une base de P_k .
- $B_k \subset B_{k+1}$.

Par exemple, $\forall l \in \mathbb{N}$, la famille des bases monomiales $\{\{z^i\}_{0 \leq i \leq k}\}_{k \geq 0}$ est hiérarchique. Dans notre cas, nous avons considéré la famille des polynômes de Legendre pour V_l de sorte qu'il puisse se décomposer sous la forme d'unions disjointes de la manière suivante :

$$V_l = \bigsqcup_{m=0}^l \tilde{V}_m \quad (4.7)$$

tel que $\text{Vect}(V_l)$ est une base de P_l^1 avec $\text{Vect}(V_l) \subset \text{Vect}(V_{l+1})$, $\forall m \in \llbracket 0, l \rrbracket, \forall f \in \tilde{V}_m, \deg(f) = m$.

2.2 Génération d'une base d'éléments finis 3D à partir d'une somme de produits tensoriels entre W_k et \tilde{V}_{n-k}

D'abord, introduisons la propriété fondamentale utilisée dans toute la suite. Soient A, B deux sous-espaces vectoriels d'un espace vectoriel E , on a alors l'égalité suivante :

$$\text{Vect}(A \cup B) = \text{Vect}(A) + \text{Vect}(B) \quad (4.8)$$

Ce résultat est bien connu et se démontre par double inclusion.

Ensuite, $\forall l \in \mathbb{N}$, décomposons P_l^1 de manière hiérarchique à partir de la famille des bases monomiales en écrivant sous forme de sommes directes :

$$P_l^1 = \bigoplus_{i=0}^l \tilde{P}_i^1 \quad \text{avec } \tilde{P}_i^1 = \text{Vect}(\{z^i\}) \quad (4.9)$$

Grâce à cette décomposition, on peut alors montrer que :

$$P_k^3 = \sum_{m=0}^k P_m^2 \otimes \tilde{P}_{k-m}^1 \quad (4.10)$$

Prouvons alors l'Equation (4.10)

Montrons que $\sum_{m=0}^k P_m^2 \otimes \tilde{P}_{k-m}^1 = \text{Vect}(\{x^\alpha y^\beta z^\gamma\}_{\alpha+\beta+\gamma \leq k})$.

On a :

$$\sum_{m=0}^k P_m^2 \otimes \tilde{P}_{k-m}^1 = \sum_{m=0}^k \text{Vect}(\{x^\alpha y^\beta z^{k-m}\}_{\alpha+\beta \leq m})$$

De plus :

$$\begin{aligned} \sum_{m=0}^k \text{Vect}(\{x^\alpha y^\beta z^{k-m}\}_{\alpha+\beta \leq m}) &= \text{Vect}\left(\bigcup_{m=0}^k \{x^\alpha y^\beta z^{k-m}\}_{\alpha+\beta \leq m}\right) \\ &= \text{Vect}(\{x^\alpha y^\beta z^\gamma\}_{\alpha+\beta+\gamma \leq k}) \end{aligned}$$

Car :

$$\begin{aligned} \{x^\alpha y^\beta z^\gamma\}_{\alpha+\beta+\gamma \leq k} &= \{x^\alpha y^\beta z^\gamma\}_{\substack{\alpha+\beta \leq k-m \\ \gamma \leq m}} \\ &= \bigcup_{m=0}^k \{x^\alpha y^\beta z^{k-m}\}_{\alpha+\beta \leq m} \end{aligned}$$

Par définition de $P_k^3 = \text{Vect}(\{x^\alpha y^\beta z^\gamma\}_{\alpha+\beta+\gamma \leq k})$, ainsi, l'Equation (4.10) est prouvée.

À partir de l'Equation (4.10), le résultat suivant peut alors être obtenu :

$$P_k^3 \subset \sum_{m=0}^k \text{Vect}(W_m \otimes \tilde{V}_{k-m}) \quad (4.11)$$

Preuve de l'Equation (4.11)

Il est important de noter que $\tilde{P}_i \not\subset \text{Vect}(\tilde{V}_i)$; par conséquent, montrons que $\forall m \in \llbracket 0, k \rrbracket$,

$$P_m^2 \otimes \tilde{P}_{k-m}^1 \subset \sum_{j=0}^k \text{Vect}(W_j \otimes \tilde{V}_{k-j})$$

On a :

$$P_m^2 \otimes \tilde{P}_{k-m}^1 \subset P_m^2 \otimes \text{Vect}\left(\bigsqcup_{i=0}^{k-m} \tilde{V}_{k-m-i}\right) = \sum_{i=0}^{k-m} P_m^2 \otimes \text{Vect}(\tilde{V}_{k-m-i}) \quad (4.12)$$

Or :

$$\begin{aligned} \sum_{i=0}^{k-m} P_m^2 \otimes \text{Vect}(\tilde{V}_{k-m-i}) &\subset \sum_{i=0}^{k-m} P_{m+i}^2 \otimes \text{Vect}(\tilde{V}_{k-m-i}) \\ &= \sum_{j=m}^k P_j^2 \otimes \text{Vect}(\tilde{V}_{k-j}) \end{aligned} \quad (4.13)$$

De plus :

$$\begin{aligned} \sum_{j=m}^k P_j^2 \otimes \text{Vect}(\tilde{V}_{k-j}) &\subset \sum_{j=0}^k P_j^2 \otimes \text{Vect}(\tilde{V}_{k-j}) \\ &\subset \sum_{j=0}^k \text{Vect}(W_j) \otimes \text{Vect}(\tilde{V}_{k-j}) = \sum_{j=0}^k \text{Vect}(W_j \otimes \tilde{V}_{k-j}) \end{aligned} \quad (4.14)$$

Cela étant vrai $\forall m \in \llbracket 0, k \rrbracket$, alors $P_k^3 \subset \sum_{m=0}^k \text{Vect}(W_m \otimes \tilde{V}_{k-m})$, CQFD.

Pour que la famille de fonctions $\sum_{m=0}^n \text{Vect}(W_m^2 \otimes \tilde{V}_{n-m})$ soit une base de P_k^3 , il est nécessaire que celle-ci soit libre *i.e.* $\text{Vect}\left(\bigcap_{m=0}^k W_m \otimes \tilde{V}_{k-m}\right) = \{0\}$. Si cet espace est libre, alors l'espace d'approximation suivant peut être généré :

$$P = \bigoplus_{m=0}^k \text{Vect}(W_m \otimes \tilde{V}_{k-m}) \quad (4.15)$$

Cette base est nommée **base additive** et le solveur associé **solveur additif**.

Prouvons que $\sum_{m=0}^k \text{Vect}(W_m \otimes \tilde{V}_{k-m})$ est libre.

Pour cela, introduisons les réels $\lambda_i^m, \forall (i, m) \in [[0, N(m)]] \times [[0, k]]$ en considérant la combinaison linéaire suivante,

$$\sum_{m=0}^k \sum_{i=0}^{N(m)} \lambda_i^m w_i^m \phi_{k-m} = 0 \quad (4.16)$$

et montrons que $\forall (i, m) \in [[0, N(m)]] \times [[0, k]], \lambda_i^m = 0$:

En effet, $\forall h \in [[0, k]]$, l'Equation (4.16) induit que :

$$\begin{aligned} \langle \sum_{m=0}^k \sum_{i=0}^{N(m)} \lambda_i^m w_i^m \phi_{k-m}, \phi_{k-h} \rangle_{L^2(\kappa_{1D})} &= 0 \\ \Leftrightarrow \sum_{i=0}^{N(h)} \lambda_i^h w_i^h \|\phi_{k-h}\|_{L^2(\kappa_{1D})}^2 &= 0 \end{aligned} \quad (4.17)$$

L'Equation (4.17) est obtenue à partir des propriétés d'orthogonalité de la base V_l qui, pour rappel, correspond à la base des polynômes de Legendre.

Or W_h est une base de fonctions, par conséquent, c'est une famille libre, ainsi $\forall i \in [[1, N(h)]] \lambda_i^h = 0$. La même démarche peut s'appliquer $\forall h \in [[0, k]]$, de telle sorte que $\forall (i, m) \in [[0, N(m)]] \times [[0, k]], \lambda_i^m = 0$ et alors $\sum_{m=0}^k \text{Vect}(W_m \otimes \tilde{V}_{k-m})$ constitue une famille libre.

Dans ce cadre, l'élément fini (κ, P, Σ) suivant peut être défini :

- κ : une maille 3D correspondant à un hexagone extrudé.
- $P = \bigoplus_{m=0}^k \text{Vect}(W_m \otimes \tilde{V}_{k-m})$: espace d'approximation.
- Σ : ensemble de formes linéaires sur P .

Les fonctions de Σ se construisent de la même manière que dans le cas de la base multiplicative, les différences proviennent des termes intervenant dans le système linéaire. Également, par des arguments similaires à ceux utilisés pour la base multiplicative, il peut être prouvé que Σ est P -unisolvant.

Par ailleurs, il est intéressant de regarder le nombre de fonctions de base de P . Dans le cas où W_m est généré par les fonctions de Wachspress, cela conduit à :

$$\dim(P) = \sum_{m=0}^k \text{card}(W_m) \times \text{card}(\tilde{V}_{k-m}) = 1 + \sum_{m=1}^k 6 \frac{k(k+1)}{2} = 3k^2 + 3k + 1 \quad (4.18)$$

En revanche si W_m est constitué des fonctions polynomiales, cela implique :

$$\begin{aligned}
 \dim(P) &= \sum_{m=0}^k \text{card}(W_m) \times \text{card}(\tilde{V}_{k-m}) = \sum_{m=0}^k \frac{(k+1)(k+2)}{2} \\
 &= \sum_{m=0}^k \frac{k^2 + 3k + 2}{2} = \frac{1}{2} \left(\frac{k(k+1)(2k+1)}{6} + \frac{1}{2} 3k(k+1) + 2(k+1) \right) \quad (4.19) \\
 &= \frac{k+1}{12} (2k^2 + 10k + 12)
 \end{aligned}$$

En analysant le Tableau 4.1 illustrant le nombre de fonctions de base pour les différentes bases 3D possibles, une première comparaison entre la base multiplicative et la base additive peut être effectuée sur le cardinal de P . Par exemple, si les W_m sont générés à partir des fonctions de Wachspress, à un terme près, il y a deux fois moins d'éléments dans la base additive que dans la base multiplicative. Un constat relativement similaire est visible dans le cas où W_m est généré par des fonctions polynomiales. Également, la base additive obtenue par les fonctions de Wachspress et la base multiplicative issue des polynômes comportent approximativement le même nombre d'éléments jusqu'à l'ordre 3. Les différences les plus notables proviennent de la comparaison entre la base additive des polynômes et la base multiplicative de Wachspress où il peut y avoir jusqu'à quatre fois plus de fonctions avec cette dernière.

TABLEAU 4.1 Nombre de fonctions utilisées pour les différentes bases 3D possibles

Ordre de la base XYZ	card(P) avec la base de Wachspress		card(P) avec la base de polynômes	
	Base multiplicative	Base additive	Base multiplicative	Base additive
1	12	7	6	4
2	36	19	18	10
3	72	37	40	20
4	120	61	75	35
5	180	91	126	56

3 Synthèse

Ce chapitre propose la construction de plusieurs bases d'éléments finis 3D à partir des bases 2D obtenues précédemment. Une première base 3D a été générée par l'utilisation du produit tensoriel. Cette approche existe déjà dans la littérature, notamment pour la génération d'éléments finis sur des maillages prismatiques 3D [9, 107]. Une seconde base originale a été développée dans l'optique de couvrir au plus proche l'espace P_k^3 afin de réduire les degrés de liberté. Celle-ci correspond en réalité à la somme directe de produits tensoriels d'espace d'éléments finis 2D et 1D. Il est intéressant de noter que la base additive comporte approximativement deux fois moins de degrés de liberté que la base multiplicative. Par la suite, ces deux différents types de base vont être comparés sur un benchmark hexagonal 3D à partir de calculs de k_{eff} et de taux d'absorption.

Chapitre 5

Résolution de l'équation de transport en 3D avec différentes bases d'éléments finis

Suite aux développements théoriques effectués précédemment, l'équation de transport est maintenant résolue avec les solveurs additif et multiplicatif en trois dimensions. Premièrement, nous vérifierons que ces bases d'éléments finis sont bien adaptées au schéma de discrétisation au travers de quelques cas MMS bien choisis. Un premier cas consiste à s'assurer que les espaces d'approximation engendrent bien numériquement des espaces polynomiaux de dimension trois à un ordre k donné. Également, une deuxième vérification est réalisée avec une fonction $C^\infty(\mathcal{D})$ pour garantir que l'on obtient des ordres de convergence numériques proches de ceux théoriques avec les bases additives et multiplicatives.

Enfin, des simulations sur un cas réacteur 3D ont été réalisées au travers du benchmark de Takeda [100] déjà utilisé en 2D dans le chapitre 3. L'idée est une nouvelle fois de comparer ces bases d'éléments finis 3D à une solution de référence en analysant les résultats sur le facteur de multiplication k_{eff} et sur les taux d'absorption.

Les comparaisons sont réalisées dans deux contextes différents : le cas où W_k est l'ensemble des fonctions de Wachspress et le cas où il correspond à l'ensemble des monômes orthogonalisés.

1 Vérification des bases d'éléments finis 3D avec les MMS

De manière similaire au chapitre 3, nous nous assurons premièrement que pour les fonctions $\psi_m \in P$, l'erreur $\|\psi - \psi_m\|_{L^2(K)}$ est proche de zéro. Dans un second temps, nous considérons une solution $C^\infty(\mathcal{D})$ (où \mathcal{D} est un domaine cartésien 3D) pour vérifier que l'on retrouve bien les ordres de convergence attendus.

Aussi, pour pouvoir effectuer les vérifications suivantes, nous considérons que \mathcal{D} est maillé avec des hexagones prismatiques par morceaux¹. La démarche est alors la même que celle du chapitre 3. En effet, $\partial\mathcal{D}$ ne pouvant être maillé exclusivement avec des hexagones, nous avons introduit une nouvelle fois la notion de cellules fantômes (voir la Figure 3.4 pour rappel pour le cas en deux dimensions). Par conséquent, plutôt que de résoudre l'équation de transport sur \mathcal{D} , nous la résolvons sur K où K est l'ensemble des hexagones contenus dans \mathcal{D} (cf. l'Equation (3.16)). Les cellules fantômes permettent alors d'imposer les conditions aux limites sur ∂K .

Enfin, précisons ici que l'objectif ne consiste pas à analyser la variation des erreurs numériques en fonction de la régularité de ψ_m mais plutôt de s'assurer que la construction des bases 3D à partir d'une base 1D et de nos bases 2D vérifiées en amont sont bien adaptées à la discrétisation DG- S_n de l'équation de transport.

1.1 Solutions dans l'espace d'approximation

Trois fonctions ψ_m appartenant à P_1^3 , P_2^3 et à P_3^3 ² sont considérées. Les Tableaux 5.1 et 5.2 illustrent les erreurs numériques respectivement obtenues pour les bases multiplicatives et additives.

Conformément à la théorie, les erreurs sont proches de zéro dès que l'ordre de la base est supérieur ou égal à celui de la fonction à projeter : $\|\psi_m - \psi_h\|_{L^2(K)} \leq 5 \times 10^{-8}$. Aussi, l'erreur numérique obtenue pour la fonction $\psi_m(x, y, z) = x^2z + 3xyz + y^2z + z^3 + z^2 + 1$ est de 5.21×10^{-9} pour l'espace $P = \text{Vect}(W_2 \otimes V_3)$. Ce résultat est en accord avec la théorie car $\text{Vect}(W_2 \otimes V_3)$ engendre $P_2^2 \otimes P_3^1$.

Il est intéressant de noter que les bases additives produisent des erreurs numériques inférieures (approximativement d'un ordre de grandeur) à celles générées par les bases multiplicatives. Cela peut se justifier en considérant la projection d'une fonction $\psi_m \in P_k^3$

1. Le maillage axial n'est pas nécessairement uniforme

2. Cette troisième fonction est également dans $P_2^2 \otimes P_3^1$.

TABLEAU 5.1 Erreurs $\|\psi_m - \psi_h\|_{L^2(K)}$ pour trois fonctions ψ_m . Les erreurs sont calculées avec la base multiplicative $P = \text{Vect}(W_k \otimes V_l)$ avec $(k, l) \in \llbracket 1, 3 \rrbracket^2$

Ordre XY-Z	$N_{ddl} = N_k \times \text{card}(P)$	Fonctions ψ_m		
		$1 + x + y$	$1.25xy + xz + yz + z^2$	$x^2z + 3xyz + y^2z + z^3 + z^2 + 1$
XY1-Z1	$108 = 9 \times 12$	5.25×10^{-13}	1.18×10^3	3.65×10^4
XY1-Z2	$162 = 9 \times 18$	2.48×10^{-12}	4.79×10^1	3.75×10^3
XY1-Z3	$216 = 9 \times 24$	3.86×10^{-12}	4.79×10^1	3.73×10^3
XY2-Z1	$216 = 9 \times 24$	1.46×10^{-12}	1.18×10^3	2.92×10^4
XY2-Z2	$324 = 9 \times 36$	2.26×10^{-11}	2.43×10^{-10}	4.44×10^2
XY2-Z3	$432 = 9 \times 48$	3.17×10^{-11}	3.59×10^{-10}	5.21×10^{-9}
XY3-Z1	$324 = 9 \times 36$	1.38×10^{-11}	1.18×10^3	2.92×10^4
XY3-Z2	$486 = 9 \times 54$	1.83×10^{-10}	1.90×10^{-9}	4.43×10^2
XY3-Z3	$648 = 9 \times 72$	2.63×10^{-10}	3.02×10^{-9}	4.40×10^{-8}

TABLEAU 5.2 Erreurs $\|\psi_m - \psi_h\|_{L^2(K)}$ obtenues pour trois fonctions ψ_m avec les bases additives.

Ordre XY-Z	$N_{ddl} = N_k \times \text{card}(P)$	Fonctions ψ_m		
		$1 + x + y$	$1.25xy + xz + yz + z^2$	$x^2z + 3xyz + y^2z + z^3 + z^2 + 1$
XY1-Z1	$63 = 9 \times 7$	8.37×10^{-13}	6.28×10^2	1.18×10^4
XY2-Z2	$171 = 9 \times 19$	5.45×10^{-12}	5.63×10^{-11}	1.29×10^3
XY3-Z3	$333 = 9 \times 37$	3.28×10^{-11}	3.29×10^{-10}	5.69×10^{-9}

dans l'espace $P = \text{Vect}(W_k \otimes V_l)$ tel que P engendre P_k^3 . Par conséquent, les coefficients de ψ_m associés aux fonctions permettant de générer les monômes $x^\alpha y^\beta \notin P_k^3$ doivent être nuls³. Bien que cela soit vrai mathématiquement, ce n'est pas exactement le cas numériquement. Cela conduit alors à des erreurs plus élevées avec les bases multiplicatives qu'avec les bases additives pour les fonctions $\psi_m \in P_k^3$.

1.2 Solution $\psi_m \in C^\infty(K)$, $\psi_m \notin P$

Considérons la fonction suivante :

$$\psi_m(x, y, z) = \sin\left(\frac{2\pi x}{L_x}\right) \sin\left(\frac{2\pi y}{L_y}\right) \sin\left(\frac{2\pi z}{L_z}\right) \quad (5.1)$$

Nous voulons nous assurer ici que l'on obtient les ordres de convergence attendus en considérant l'estimateur d'erreur *a priori* de l'Equation (3.17) introduit dans le chapitre 3.

3. Dans le cas où W_k est l'ensemble des fonctions de Wachspress, il doit y avoir une simplification entre les différentes fonctions de base sur la face où la fonction dégénère en ce monôme.

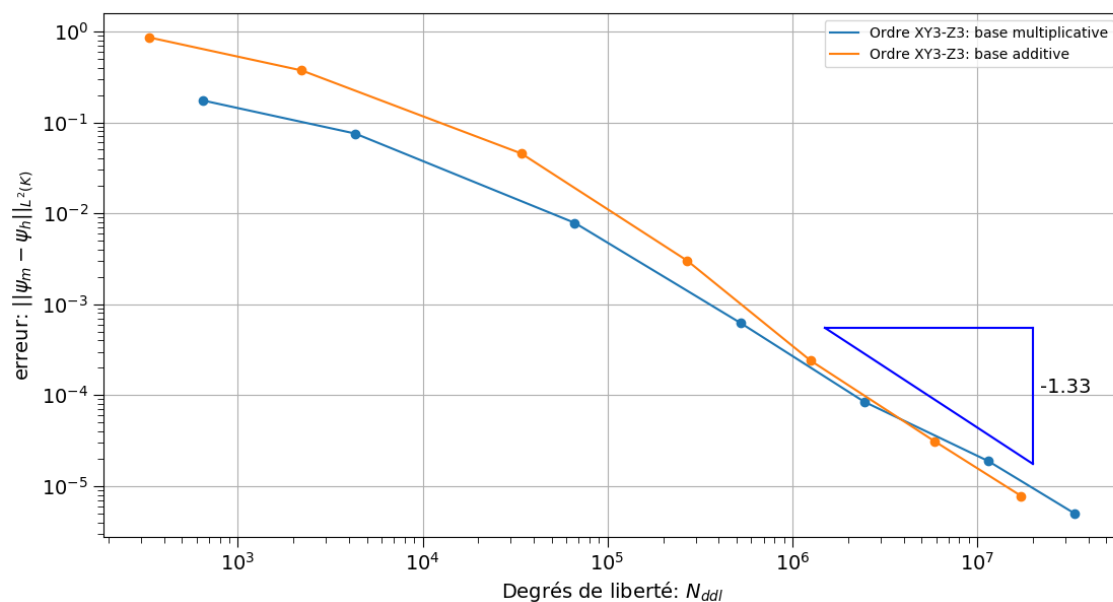


FIGURE 5.1 $e_h = \|\psi_m - \psi_h\|_{L^2(K)}$ pour la base additive et la base multiplicative à l'ordre 3

Pour cela, nous considérons les bases additive et multiplicative à l'ordre 3. Comme l'illustre la Figure 5.1, les ordres de convergence théoriques sont approximativement retrouvés. Aussi, pour le même nombre de maille, la base multiplicative produit des erreurs numériques inférieures (environ huit fois inférieures) à la base additive jusqu'à $N_{ddl} = 10^6$. À partir de ce nombre, les erreurs sont approximativement les mêmes. Ce résultat est contre-intuitif car les degrés de liberté supplémentaires offerts par la base multiplicative sont utiles pour approximer cette fonction ψ_m . Des études supplémentaires seront nécessaires pour mieux comprendre si cela est dû à des effets numériques pour un certain nombre de degrés de liberté. Une idée serait alors de refaire les mêmes calculs mais en considérant des fréquences plus faibles pour cette fonction ψ_m pour que le problème converge avec un nombre de degrés de liberté inférieure.

Grâce à ces vérifications, les bases 3D développées sont bien adaptées à notre schéma de discrétisation. Nous pouvons maintenant nous intéresser à des cas plus réalistes comme les benchmarks de réacteurs nucléaires.

2 Résultats numériques sur un cas réacteur

Pour comparer nos bases d'éléments finis, nous utilisons le quatrième modèle décrit par Takeda dans [100] (voir Figure 3.15) qui a déjà été introduit dans le chapitre 3. Nos résultats numériques porteront sur le k_{eff} et sur les taux d'absorption condensés à un groupe d'énergie. Pour obtenir ces différentes grandeurs, quatre bases d'éléments finis sont utilisées par notre solveur maquette. Deux bases additives et deux bases multiplicatives, les différences provenant de l'espace W_k . En effet, nous considérons deux cas pour l'espace W_k , soit l'ensemble des fonctions de Wachspress (HOW), soit l'ensemble des monômes orthogonalisés (POLY). Comme dans les cas traités précédemment en deux dimensions, SNATCH génère d'une part une solution de référence à un ordre élevé avec un maillage raffiné⁴. D'autre part, SNATCH produit des solutions sur un maillage non-raffiné où chaque hexagone est découpé en trois losanges pour les ordres 1 à 5 que l'on note SNS1. Ces solutions de SNATCH sont utilisées pour comparer les performances de nos schémas numériques. Soulignons qu'une base de SNATCH à un ordre k donné engendre l'espace P_k^3 et ne prend pas en compte des ordres mixtes dans le plan radial et pour l'axe z .

2.1 Description du benchmark

Pour ce modèle, Takeda décrit trois configurations possibles. Dans ces travaux, nous considérons le troisième cas correspondant aux barres de contrôles totalement insérées. Les données concernant les matrices de diffusion et les sections efficaces sont disponibles dans [100]. La partie axiale de ce benchmark se décompose en sept couches : $\{[0.0, 45], [45, 65], [65, 95], [95, 125], [125, 145], [145, 155], [155, 190]\}$. Pour ces couches axiales, seuls les matériaux constituant les trois premières couronnes autour de l'hexagone central varient. Par conséquent, le maillage correspondant à ce benchmark peut être généré en connaissant les matériaux décrits par plan selon le segment $[A, A']$ (cf. Figure 3.15) comme illustré dans [100] et le Tableau 5.3.

Également, ce benchmark comporte quatre groupes d'énergie et l'ensemble des calculs se sont effectués pour une quadrature angulaire produit de type Gauss-Legendre d'ordre 4 pour l'angle polaire et Gauss-Chebyshev d'ordre 3 pour l'angle azimutal ; ce qui conduit à 144 directions. .

4. La solution est obtenue en maillant chaque hexagone en $3 \times 2^3 = 24$ losanges et en considérant une base d'ordre 5.

TABLEAU 5.3 Répartition des matériaux par plan selon une diagonale du cœur (cf. la Figure 3.15)

Hauteur (cm)	Répartition des matériaux par plan selon la diagonale du cœur														
								Hexagone central							
190 155 145	10	9	8	7	1	1	1	1	1	1	1	7	8	9	10
145 125	10	9	8	7	3	3	2	2	2	3	3	7	8	9	10
125 95 65	10	9	8	7	6	5	4	4	4	5	6	7	8	9	10
65 45	10	9	8	7	3	3	2	2	2	3	3	7	8	9	10
45 0	10	9	8	7	1	1	1	1	1	1	1	7	8	9	10

1 : Acier - 2 : Couverture axiale - 3 : Réflecteur axial
4 : Zone de test - 5 : Zone nourricière avec modérateur - 6 : Zone nourricière sans modérateur
7 : Réflecteur sans modérateur - 8 : Réflecteur avec modérateur - 9 : Réflecteur KNK-1
10 : Zone de sodium/acier

2.2 Résultats numériques

Comme expliqué en début de section, l'analyse est réalisée à l'aide de notre solveur maquette et du solveur SNATCH. La solution de référence produit un $k_{eff} = 0.879836$ et des taux d'absorption de référence. Ces taux sont illustrés en Annexe E dans les Figures E.1 et E.2.

Aussi, nous considérons les sept plans introduits dans le Tableau 5.3 comme maillage axial. Nous réalisons alors sur ce maillage 3D des calculs de k_{eff} et de taux d'absorption pour les ordres 1 à 5 avec les bases additives et multiplicatives. Pour ces dernières, les ordres radiaux k et axiaux l , tels que $(k, l) \in \llbracket 1, 5 \rrbracket^2$ sont considérés. Notons MP et AD respectivement les bases multiplicatives et additives. Précisons que les critères d'arrêt pour les itérations internes, multigroupes et de la puissance sont de 1×10^{-5} .

Également, nous reprenons les mêmes notations qu'auparavant *i.e.* les calculs sont caractérisés par le nombre de degrés de liberté N_{ddl} associé à la discrétisation spatiale où $N_{ddl} = N_{\kappa} \times N(k)$, avec N_{κ} le nombre de cellules et $N(k)$ le nombre de fonctions de base par maille.

TABLEAU 5.4 Comparaison du Δk_{eff} entre la solution de référence et SNS1 ainsi que les bases additives.

Ordre	SNS1		HOW-AD		POLY-AD	
	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]
1	28392 = 3549 × 8	1244	8281 = 1183 × 7	2780	4732 = 1183 × 4	3314
2	95823 = 3549 × 27	16.4	22477 = 1183 × 19	114	11830 = 1183 × 10	296
3	227136 = 3549 × 64	6.1	43771 = 1183 × 37	36.4	23660 = 1183 × 20	59.9
4	443625 = 3549 × 125	1.7	72163 = 1183 × 61	-7.3	41405 = 1183 × 35	-5.19
5	766584 = 3549 × 216	0.9	107653 = 1183 × 91	-14.9	66248 = 1183 × 56	-8.63

TABLEAU 5.5 Comparaison du Δk_{eff} pour les bases multiplicatives avec $P = \text{Vect}(W_k \otimes V_l)$ où $(k, l) \in \llbracket 1, 2 \rrbracket \times \llbracket 1, 5 \rrbracket$

Ordre XY-Z	HOW-MP		POLY-MP	
	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]
XY1-Z1	14196 = 1183 × 12	2040	7098 = 1183 × 6	2520
XY1-Z2	21294 = 1183 × 18	1490	10647 = 1183 × 9	1970
XY1-Z3	28392 = 1183 × 24	1480	14196 = 1183 × 12	1970
XY1-Z4	35490 = 1183 × 30	1480	17745 = 1183 × 15	1970
XY1-Z5	42588 = 1183 × 36	1480	21294 = 1183 × 18	1970
XY2-Z1	28392 = 1183 × 24	588	14196 = 1183 × 12	745
XY2-Z2	42588 = 1183 × 36	18.8	21294 = 1183 × 18	179
XY2-Z3	56784 = 1183 × 48	14.3	28392 = 1183 × 24	176
XY2-Z4	70980 = 1183 × 60	13.1	35490 = 1183 × 30	176
XY2-Z5	85176 = 1183 × 72	13.2	42588 = 1183 × 36	173

TABLEAU 5.6 Comparaison du Δk_{eff} pour les bases multiplicatives avec $P = \text{Vect}(W_k \otimes V_k)$ où $k \in \llbracket 1, 5 \rrbracket$

Ordre XY-Z	HOW-MP		POLY-MP	
	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]	$N_{ddl} = N_{\kappa} \times N(k)$	Δk_{eff} [pcm]
XY1-Z1	14196 = 1183 × 12	2040	7098 = 1183 × 6	2520
XY2-Z2	42588 = 1183 × 36	18.7	21294 = 1183 × 18	179
XY3-Z3	85176 = 1183 × 72	27.5	47320 = 1183 × 40	42.9
XY4-Z4	141960 = 1183 × 120	-9.7	88725 = 1183 × 75	-7.3
XY5-Z5	212940 = 1183 × 180	3	149058 = 1183 × 126	9.6

TABLEAU 5.7 Erreurs max et rms sur les taux d'absorption entre la solution de référence et SNS1 ainsi que les bases additives.

Ordre	SNS1			HOW-AD			POLY-AD		
	N_{ddl}	max[-]	rms[-]	N_{ddl}	max[-]	rms[-]	N_{ddl}	max[-]	rms[-]
1	28392	1.8×10^{-1}	6.8×10^{-2}	8281	2.1×10^{-1}	9.0×10^{-2}	4732	3.0×10^{-1}	1.2×10^{-1}
2	95823	2.0×10^{-3}	2.3×10^{-3}	22477	1.2×10^{-2}	1.3×10^{-2}	11830	3.1×10^{-2}	1.5×10^{-2}
3	227136	5.7×10^{-4}	5.9×10^{-4}	43771	2.3×10^{-3}	1.8×10^{-3}	23660	4.5×10^{-3}	3.3×10^{-3}
4	443625	2.0×10^{-4}	2.1×10^{-4}	72163	4.3×10^{-4}	8.6×10^{-4}	41405	6.7×10^{-4}	6.1×10^{-4}
5	766584	8.5×10^{-5}	1.2×10^{-4}	107653	3.7×10^{-2}	9.1×10^{-3}	66248	3.8×10^{-2}	1×10^{-3}

TABLEAU 5.8 Erreurs max et rms sur les taux d'absorption pour les bases multiplicatives avec avec $P = \text{Vect}(W_k \otimes V_l)$ où $(k, l) \in \llbracket 1, 2 \rrbracket \times \llbracket 1, 5 \rrbracket$

Ordre XY-Z	HOW-MP			POLY-MP		
	N_{ddl}	max[-]	rms[-]	N_{ddl}	max[-]	rms[-]
XY1-Z1	14196	2.1×10^{-1}	7.2×10^{-2}	7098	2.9×10^{-1}	1.0×10^{-1}
XY1-Z2	21294	6.3×10^{-2}	2.7×10^{-2}	10647	1.6×10^{-1}	6.1×10^{-2}
XY1-Z3	28392	6.3×10^{-2}	2.7×10^{-2}	14196	1.6×10^{-1}	6.1×10^{-2}
XY1-Z4	35490	6.3×10^{-2}	2.7×10^{-2}	17745	1.6×10^{-1}	6.1×10^{-2}
XY1-Z5	42588	6.3×10^{-2}	2.7×10^{-2}	21294	1.6×10^{-1}	6.1×10^{-2}
XY2-Z1	28392	1.7×10^{-1}	7.1×10^{-2}	14196	1.8×10^{-1}	6.9×10^{-2}
XY2-Z2	42588	1.2×10^{-2}	3.2×10^{-3}	21294	1.1×10^{-2}	6.8×10^{-3}
XY2-Z3	56784	1.2×10^{-2}	3.1×10^{-3}	28392	1.0×10^{-2}	6.4×10^{-3}
XY2-Z4	70980	1.2×10^{-2}	2.9×10^{-3}	35490	1.0×10^{-2}	6.5×10^{-3}
XY2-Z5	85176	9.0×10^{-3}	2.2×10^{-3}	42588	1.0×10^{-2}	6.1×10^{-3}

TABLEAU 5.9 Erreurs max et rms sur les taux d'absorption pour les bases multiplicatives avec $P = \text{Vect}(W_k \otimes V_k)$ où $k \in \llbracket 1, 5 \rrbracket$

Ordre XY-Z	HOW-MP			POLY-MP		
	N_{ddl}	max[-]	rms[-]	N_{ddl}	max[-]	rms[-]
XY1-Z1	14196	2.1×10^{-1}	7.2×10^{-2}	7098	2.9×10^{-1}	1.0×10^{-1}
XY2-Z2	42588	1.2×10^{-2}	3.2×10^{-3}	21294	1.1×10^{-2}	6.8×10^{-3}
XY3-Z3	85176	1.1×10^{-2}	3.0×10^{-3}	47320	2.0×10^{-3}	4.2×10^{-3}
XY4-Z4	141960	1.2×10^{-2}	2.7×10^{-3}	88725	1.0×10^{-3}	1.6×10^{-3}
XY5-Z5	212940	1.5×10^{-2}	3.5×10^{-3}	149058	1.8×10^{-2}	4.0×10^{-3}

Tendance globale entre SNS1 et nos bases

Les Tableaux 5.4, 5.5 et 5.6 présentent les écarts sur le k_{eff} tandis que les Tableaux 5.7, 5.8, 5.9 donnent les écarts sur les taux d'absorption (nous analysons ici l'écart maximal ainsi que la moyenne quadratique des écarts (rms)).

Sur ce benchmark, l'utilisation de nos bases additives et multiplicatives à un ordre 3 ou 4 – en particulier POLY-AD et HOW-AD – se révèlent très intéressante pour réduire le nombre

de degrés de liberté par rapport à SNATCH pour une précision visée de l'ordre de 10 pcm sur le k_{eff} et de 10^{-3} sur les taux d'absorption. En particulier, POLY-AD à l'ordre 4 permet de diviser par cinq le nombre de degrés de liberté par rapport à l'ordre 3 de SNS1 pour des résultats du même ordre de grandeur.

En revanche, pour une précision plus grande, une discrétisation des hexagones est nécessaire. En effet, pour le cas de l'ordre 5, SNS1 produit un Δk_{eff} de 0.9 là où par exemple POLY-MP et POLY-AD induisent des Δk_{eff} respectifs de 9.6 et -8.6 cf. les lignes en turquoise des Tableaux 5.4 et 5.6. Le Δk_{eff} est alors presque dix fois inférieur avec SNS1 qu'avec nos bases. Un bilan similaire peut être réalisé sur les taux. Effectivement, les erreurs max et rms produites par SNS1 sont en moyenne d'un ordre de grandeur inférieur à celles des bases AD et MP. En particulier à l'ordre 5, comme mis en évidence sur les lignes en turquoise des Tableaux 5.7 et 5.9, l'erreur max de SNS1 est très inférieure à celles obtenues par AD et MP mais ces erreurs sont localisées sur des zones de faibles puissances telles que les réflecteurs ou l'acier. Toujours pour cet ordre, l'erreur rms de SNS1 est d'un ordre de grandeur inférieure à celles de nos base.

Au vu des différents résultats obtenus, favoriser uniquement le raffinement en ordre ne semble pas être la meilleure stratégie pour ce benchmark qui reste un problème difficile avec des zones d'hétérogénéités matérielles très fortes. En effet, à partir de l'ordre 4, les différences sont significatives entre SNS1 et nos bases. Par conséquent, le raffinement en h produit des solutions plus proches de la solution de la référence que le raffinement en p . Takeda dans [100] avait déjà évoqué ce problème où il précise que les solutions obtenues par un solveur S_n (S_8) n'étaient pas précises comparées aux solutions de référence et qu'une piste d'amélioration serait d'utiliser un maillage plus fin.

Comparaisons entre les bases multiplicatives

Si nous nous intéressons aux Tableaux 5.5 et 5.8, il est intéressant de souligner l'importance de l'augmentation de l'ordre 1 à l'ordre 2 pour la variable axiale. En effet, par exemple dans le cas de HOW-MP pour le Δk_{eff} , ce dernier est de 588 à l'ordre XY2-Z1 alors qu'il est de 18.8 pour XY2-Z2 (cf. ligne en rouge sur le Tableau 5.5). Ce résultat est également mis en évidence en rouge sur le Tableau 5.8 pour les erreurs rms et max sur les taux d'absorption. En effet, ces dernières diminuent d'un ordre de grandeur avec POLY-MP lors de l'augmentation de l'ordre XY2-Z1 à l'ordre XY2-Z2.

Toutefois, au-delà de l'ordre 2 voire 3, l'augmentation de l'ordre de la base axiale n'a plus d'impact significatif sur les résultats. Ce constat est illustré en bleu dans les Tableaux

5.5 et 5.8 lors du passage de l'ordre XY2-Z3 à XY2-Z4. Pour ces ordres, le Δk_{eff} diminue seulement d'un pcm pour la base HOW-MP et les erreurs max et rms ne varient que d'un facteur inférieur à 1.1 (voire augmente dans le cas de l'erreur rms avec POLY-MP). Par conséquent, l'erreur n'est plus dominée par la discrétisation axiale mais la discrétisation radiale. Ce résultat est cohérent car comme précisé dans [23], le flux axial peut être approché par des fonctions d'ordre faible.

Aussi, le régime asymptotique en k semble être atteint pour ce benchmark aux ordres 3-4⁵, affectant alors la convergence sur le k_{eff} et sur les taux d'absorption. Ainsi, des oscillations vis-à-vis de la précision sur les grandeurs d'intérêt sont alors visibles comme en témoigne l'augmentation des erreurs max et rms ainsi que du Δk_{eff} lors du passage de l'ordre XY4-Z4 à l'ordre XY5-Z5 (cf. les lignes en mauves et turquoises du Tableau 5.9).

Enfin, soulignons que les différences entre HOW-MP à POLY-MP sont principalement marquées aux ordres 1 et 2 pour les variables radiales de par un nombre de degrés de libertés plus important pour HOW. Par exemple, à l'ordre XY2-Z2, il y a Δk_{eff} de 179 avec POLY-MP contre un Δk_{eff} de 18.8 avec HOW-MP. Un résultat analogue est visible sur les taux où HOW-MP conduit à une erreur rms deux fois inférieure à celle de POLY-MP. Toutefois, à partir de l'ordre XY3-Z3, les différences ne sont plus significatives entre les deux bases, le bilan est alors similaire à celui réalisé en deux dimensions.

Comparaisons entre les bases additives

L'analyse réalisée ici est quasi identique à celle faite précédemment pour les bases multiplicatives. Par exemple, le Δk_{eff} obtenu avec HOW-AD est inférieur à celui de POLY-AD de plusieurs dizaines de pcm d'écart jusqu'à l'ordre 3. De même, les erreurs sur les taux d'absorption sont très similaires entre les deux bases, seul l'ordre 3 conduit à des erreurs inférieures d'un facteur deux en faveur de HOW-AD comparé à POLY-MP cf. la ligne en bleu sur le Tableau 5.7.

D'autre part, pour les ordres 3 et 4, comme expliqué auparavant, le régime asymptotique en k semble alors être atteint. Par conséquent, des variations sont visibles sur les taux ainsi que sur le k_{eff} . Elles sont particulièrement marquées ici en comparaison des bases multiplicatives, avec en particulier, les erreurs max et rms qui augmentent de plus d'un ordre de grandeur comme illustré sur le Tableau 5.7 avec les lignes mauve et turquoise.

5. Pour les variables radiales et axiales.

De manière générale, comme dans le cas des bases multiplicatives et dans les cas 2D, les différences entre HOW-AD et POLY-AD ne sont pas extrêmement prononcées.

Comparaisons entre les bases additives et multiplicatives

Si l'on compare maintenant les bases AD et MP, notons que pour les ordres $k \in \{1, 2, 3, 5\}$, les Δk_{eff} sont supérieurs avec les bases AD qu'avec les bases MP. En revanche, pour les erreurs max-rms sur les taux d'absorption, ce constat est inversé pour les ordres 3-4. On notera en particulier que pour l'ordre 4 (cf. lignes en mauve dans les Tableaux 5.9 et 5.7), AD produit des erreurs rms deux fois inférieures à celles de MP. Toujours pour cet ordre, dans le cas de l'erreur max, il y a presque un ordre grandeur entre HOW-MP et HOW-AD en faveur de ce dernier.

De manière générale, il est important de souligner le faible nombre de degrés de liberté des bases additives en comparaison des bases multiplicatives. Par exemple à l'ordre 4, POLY-AD nécessite trois fois moins de degrés de liberté que HOW-MP et conduit à un Δk_{eff} inférieur à ce dernier de presque 5 pcm. Ce constat conforte le choix d'utiliser l'option additive de APOLLO3[®]/MINARET plutôt que l'option multiplicative [2]. En effet, les degrés de liberté supplémentaires de cette dernière et par extension ceux des bases multiplicatives ne contribuent pas de manière significative à la solution du problème.

Également, une première discussion peut être ouverte sur les performances. Bien que les temps de calculs soient plus courts pour les espaces utilisant les bases de monômes orthogonalisées plutôt que les fonctions de Wachspress (du fait d'un nombre de degré de liberté plus faible à ordre égal), les bases POLY-AD et POLY-MP nécessitent plus de balayages⁶ en moyenne que les bases HOW-AD et HOW-MP. Typiquement, POLY-MP et POLY-AD effectuent respectivement (en moyenne) plus de 1600 et 2000 balayages là où ce nombre pour les bases HOW-MP et HOW-MP se situe autour de 1400. Pour mieux comprendre ces observations, il serait intéressant d'approfondir l'analyse de la convergence itérative de chacun de ces schémas numériques.

6. Le balayage correspond au nombre de fois où l'équation de transport a été résolue en angle-espace. Il totalise le nombre de résolutions effectuées pour les itérations flux-source.

3 Conclusion et perspectives

Nous avons vérifié au travers des MMS que nos bases engendrent l'espace P_k^3 numériquement. Aussi dans le cas d'une solution $\psi_m \in C^\infty(\mathcal{D})$, $\psi_m \notin P$, les ordres de convergence attendus ont été obtenus. Par conséquent, ces bases peuvent servir pour la résolution de l'équation de transport sur des maillages hexagonaux extrudés.

Ainsi, des calculs de neutronique portant sur le k_{eff} et sur les taux d'absorption ont été réalisés. On notera que pour le benchmark étudié dans ce chapitre, l'utilisation d'une discrétisation en espace s'avère particulièrement efficace pour réduire les erreurs d'approximation comme expliqué dans [100].

Cependant, il est important de souligner que pour trois fois moins de degrés de liberté, les résultats de POLY-AD à l'ordre 4 sont presque équivalents à ceux de SNS1 à l'ordre 3. Aussi, il a été observé qu'aux ordres 3-4, les bases additives conduisaient à des erreurs inférieures aux bases multiplicatives pour deux à trois fois moins de degré de liberté⁷. Par conséquent, ces dernières sont très intéressantes dans une logique de compromis entre temps de calcul et précision. Ces résultats rejoignent le postulat de départ où les utilisateurs de APOLLO3[®]/MINARET favorisent l'option additive.

De manière générale, l'objectif de départ est atteint *i.e.* développer une discrétisation capable de traiter l'hexagone afin d'obtenir un compromis entre les performances et la précision.

En perspective, des analyses plus approfondies seraient nécessaires pour comprendre les différences sur le nombre de balayages entre le cas où W_k est la base des fonctions de Wachspress et le cas où il correspond à la base des monômes orthogonalisée.

7. En fonction de si W_k est constitué des monômes orthogonalisés ou des fonctions de Wachspress.

Conclusions et perspectives

Conclusions

Dans le contexte de la simulation multiphysique des accidents graves des RNR-Na, les temps de calcul sont souvent dominés par la résolution de l'équation de transport pour la neutronique. C'est particulièrement prononcé dans la phase dite "primaire" de l'accident où la géométrie du cœur composée d'assemblages hexagonaux reste intègre et où une méthode dite "multi-canal" peut être utilisée pour la thermohydraulique. Afin de réduire le coût de la neutronique tout en conservant une certaine précision, nous avons cherché à améliorer la discrétisation spatiale associée. Cela nous a conduit à développer et comparer des bases d'éléments finis d'ordre élevé adaptées à l'hexagone dans le cadre d'une discrétisation par une méthode de Galerkin discontinue amont (DG).

Parmi les bases possibles, nous nous sommes d'abord intéressés à des bases nodales en partant des travaux de Gout qui avait proposé des bases d'éléments finis de Wachspress jusqu'à l'ordre 3 pour l'hexagone régulier. Également, de par notre discrétisation spatiale par une méthode de Galerkin discontinue, nous avons considéré des bases de monômes orthogonalisées, ces dernières ayant déjà été utilisées pour la résolution de l'équation de transport sur des maillages non structurés et courbés.

Pour ce qui est des bases de Wachspress, nous avons proposé et développé une méthode originale pour la génération de bases de fonctions de Wachspress pour un polygone avec m côtés à un ordre k tel que $k < m$. A partir des propriétés désirées pour ces fonctions de bases nodales (projection exacte des monômes jusqu'à un certain ordre, interpolation lagrangienne), l'algorithme proposé construit un système linéaire via le calcul symbolique dont les inconnues sont les coefficients de fonctions polynomiales qui apparaissent au numérateur des fonctions des éléments finis rationnels considérés. Ce système linéaire est, dans le cas général, sous-déterminé, conduisant alors à une infinité de solutions. Pour obtenir une solution particulière, la matrice augmentée du système est transformée en sa matrice échelonnée réduite de telle

sorte à déduire les variables dépendantes et indépendantes. En fixant alors ces dernières, différentes solutions peuvent être obtenues. Nous avons fait le choix arbitraire de fixer les variables indépendantes à zéro. Ces choix ont une incidence sur les erreurs d'approximation, ce résultat a été notamment observé en comparant les erreurs de projections et de contraintes entre la base de Gout et la nôtre à l'ordre 3 pour l'hexagone régulier.

Les bases développées ont alors servi pour la résolution d'EDP par une méthode d'éléments finis sur des maillages hexagonaux dans le cadre d'une maquette logicielle développée en python. Une première étape de vérification s'est effectuée par la méthode des solutions manufacturées en considérant des solutions $\psi_m \in C^\infty(K)$, $\psi_m \notin P$ afin d'évaluer les ordres de convergence obtenus numériquement⁸ en régime asymptotique. Sur une équation de Poisson (discrétisée par une méthode d'éléments finis continus) d'abord, puis sur l'équation de transport, les ordres théoriques ont bien été retrouvés. Ensuite, des premières comparaisons entre bases de Wachspress et bases de monômes orthogonalisées, au sens des erreurs d'approximation, ont été menées via, à nouveau, la méthode des solutions manufacturées à partir de solutions présentant des régularités similaires à la régularité globale des solutions de l'équation de transport en neutronique. Il a été constaté que les bases de Wachspress produisent des erreurs bien inférieures à celles des bases de monômes orthogonalisées pour tous les ordres. En particulier, moins la solution est régulière, plus les bases de Wachspress présentent une erreur moindre que les bases de monômes orthogonalisées.

Pour approfondir les comparaisons et répondre à l'objectif initial de cette thèse, deux benchmarks de cœurs de réacteurs en deux dimensions ont été calculés à l'aide de nos bases et comparés en termes de k_{eff} et de taux d'absorption à des solutions produites par les solveurs APOLLO3[®]/MINARET et SNATCH. Nous avons notamment pu observer que, pour un nombre de degrés de liberté équivalent, les erreurs d'approximation sont inférieures avec les bases développées sur l'hexagone en comparaison de celles utilisant un sous-maillage. Par exemple, pour le benchmark d'Hébert, pour un nombre de degrés de liberté légèrement supérieur, la solution de MINARET à l'ordre 1 découpant un hexagone en six triangles conduit à des erreurs plus de dix fois supérieures sur les quantités d'intérêt que la solution obtenue par la base de monômes orthogonalisée à l'ordre 4. De manière générale, pour une précision sur le k_{eff} (resp. les taux d'absorption) autour de 10 pcm (resp. entre 10^{-3} et 10^{-4}), les bases développées sur l'hexagone permettent d'améliorer significativement les performances. Par ailleurs, le gain en précision associée aux bases de Wachspress par rapport aux bases de monômes orthogonalisées n'est pas toujours intéressant vis-à-vis de l'accroissement associé du nombre de degrés de liberté.

8. Ici, à partir de l'erreur en norme $L^2(K)$ entre la solution analytique et la solution approchée.

L'extension en trois dimensions a constitué la suite de ces travaux. Dans ce cadre, un maillage prismatique a été considéré et deux bases d'éléments finis ont été développées. Une première base nommée "base multiplicative" correspond au produit tensoriel entre une base radiale (2D) à un ordre k et une base axiale (1D) à un ordre l . Cette base est un choix classique dans la littérature ; c'est, par exemple, une des options proposées par le solveur APOLLO3[®]/MINARET. Toutefois, pour des ordres $k = l$ donnés, l'espace engendré est l'espace des polynômes de degrés partiels $\leq k$ et non pas seulement P_k . En neutronique, il est constaté que cette approche n'offre pas forcément le meilleur compromis entre précision et coût de calcul. En particulier, en pratique, les utilisateurs de APOLLO3[®]/MINARET favorisent l'option "additive" de ce dernier qui correspond à une base 3D partielle construite par la somme d'une base radiale d'ordre k et d'une base axiale d'ordre l . Cette approche nous a conduit à développer une méthode de construction de bases 3D originale que nous avons également nommée "base additive" engendrant "au plus proche" l'espace P_k avec le moins de degrés de liberté possible.

Nous avons alors implémenté ces bases et étendu notre maquette logicielle au 3D. À nouveau, la vérification s'est faite par la méthode des solutions manufacturées. Ensuite, un benchmark 3D d'un cœur de réacteur a été calculé et les performances de différentes bases ont été évaluées par comparaison avec le solveur SNATCH (et un découpage des hexagones en trois losanges). Pour ce benchmark assez sévère vis-à-vis de la discrétisation spatiale, les conclusions des benchmarks 2D sont retrouvées et nos bases permettent d'améliorer significativement les performances pour une précision sur le k_{eff} (resp. les taux d'absorption) autour de 10 pcm (resp. entre 10^{-3} et 10^{-4}). Notons tout de même que pour une précision plus grande, sur ce benchmark, un sous-maillage de l'hexagone devient nécessaire. Par ailleurs, on observe que les bases additives offrent un gain accru vis-à-vis des bases multiplicatives. Ainsi, au final pour trois fois moins de degrés de liberté, les résultats de la base additive à l'ordre 4 avec les monômes orthogonalisées sont presque équivalents à ceux de SNATCH à l'ordre 3.

Perspectives

Ce travail sur les bases de Wachspress et les bases de monômes orthogonalisées a permis d'aboutir à une amélioration du compromis entre précision et temps de calcul pour des benchmarks de cœurs hexagonaux. S'en dégagent aussi des perspectives à plus ou moins long terme que nous allons maintenant présenter.

D'abord, l'algorithme de génération de bases de Wachspress pourrait être amélioré. Pour cela, une idée serait de diminuer le conditionnement de la matrice A_d afin de calculer les variables libres. À cette fin, une démarche possible est d'utiliser un préconditionneur pour la matrice A_d associé à une précision accrue pour en déduire les variables libres. Une autre possibilité serait d'ajouter de nouvelles équations⁹ lors de la construction du système ou de le générer à partir d'une base de monômes orthogonalisée. En conséquence, cela pourrait conduire à un meilleur conditionnement de la matrice A et par extension à celui de la matrice A_d . En revanche, si un trop grand nombre d'équations est rajouté, le système deviendrait éventuellement surdéterminé. Dans ce cadre, un processus supplémentaire pourrait être rajouté. Il consisterait alors en la minimisation d'une fonctionnelle sous contraintes¹⁰.

Ensuite, toujours dans la ligne directe de nos travaux, il serait intéressant d'étudier les variations de la constante C intervenant dans l'estimateur *a priori* cf. l'Equation (3.17) en fonction de la régularité de la solution et de la base choisie. Cette étude commencerait par des tests numériques via la méthode des solutions manufacturées avec des solutions analytiques présentant différentes régularités et pour différentes bases. Cela permettrait de préciser les tendances qu'il faudrait alors essayer de confirmer avec un résultat théorique à établir par la suite. Cela pourrait potentiellement offrir un critère d'optimisation du choix des variables libres pour les bases de Wachspress.

De manière plus générale en lien avec notre contexte applicatif, d'autres perspectives s'ouvrent également. Par exemple, afin de réduire l'erreur de manière optimale vis-à-vis du coût de calcul, il serait intéressant de mettre en place un schéma au raffinement adaptatif en ordre à partir d'un estimateur d'erreur *a posteriori* comme ceux décrits dans [41, 42] pour pouvoir augmenter localement l'ordre des bases. En particulier en trois dimensions, l'utilisation de la base multiplicative permettrait de distinguer les composantes radiale et axiale ce qui conduirait à un raffinement anisotrope. Il est important de souligner que le raffinement en ordre est plus aisé avec la base de monômes orthogonalisée de par son caractère hiérarchique. Une procédure décrite par [79] permet de transformer des coordonnées barycentriques généralisées en des bases hiérarchiques d'ordre quelconque sur des maillages polygonaux. Il pourrait être intéressant d'appliquer ces travaux aux fonctions de Wachspress¹¹.

Bien sûr, comme on l'a vu au travers des benchmarks simulés, augmenter l'ordre de la base peut ne pas être suffisant (ou optimal) lorsque l'on cherche une plus grande précision. Dans ce cas, il serait pertinent de s'autoriser à redécouper certains hexagones. On pour-

9. Comme des contraintes de symétrie.

10. Typiquement appliquer un opérateur de rotation à l'ensemble des fonctions de base moins sa projection dans la base, avec pour contraintes des propriétés de Lagrange et les projections des monômes.

11. Ou à d'autres GBC.

rait donc chercher à développer un solveur traitant des maillages mixtes où nos bases sur l'hexagone seraient utilisées de manière conjointe avec des bases polynomiales classiques sur des triangles ou quadrilatères. Dans ce cadre, on pourrait naturellement s'intéresser à des méthodes de raffinement *hp*.

Aussi, pour gagner en précision sur le calcul du flux, une démarche serait d'adapter les travaux de Mirzaee [76] aux maillages hexagonaux. Cet article propose d'appliquer un filtre sur une solution approchée d'une équation linéaire hyperbolique par une méthode de Galerkin discontinue sur des maillages triangulaires afin d'augmenter sa régularité. De cette manière, les ordres de convergence augmentent et donc, dans le cas d'une solution exacte continue, conduisent à des erreurs d'approximation amoindries entre la solution exacte et la solution approchée.

Bien que ces perspectives se soient focalisées sur le schéma de discrétisation spatial, qui est l'objet du travail de cette thèse, il existe d'autres axes pour améliorer le compromis performance - précision. Entre autres, le parallélisme et des méthodes d'accélération sur les itérations entre flux et sources¹² constituent des directions complémentaires afin de réduire les temps de calcul. Dans ce contexte, des études supplémentaires sont également requises pour comprendre l'impact (observé dans nos simulations) du choix de la base d'éléments finis sur la vitesse de convergence de ces itérations flux-source.

12. Comme avec la méthode d'accélération synthétique de la diffusion [105].

Bibliographie

- [1] Abu-Shumays, I. K. (2001). Angular quadratures for improved transport computations. *Transport Theory and Statistical Physics*, 30 :169–204.
- [2] Ammar, K. (2014). *Conception multi-physique et multi-objectif des cœurs de RNR-Na hétérogènes : développement d'une méthode d'optimisation sous incertitudes*. PhD thesis, Université Paris Sud-Paris XI.
- [3] Anisimov, D. (2017). *Analysis and new constructions of generalized barycentric coordinates in 2D*. PhD thesis, Università della Svizzera italiana.
- [4] Apprato, D., Arcangeli, R., and Gout, J. (1979a). Rational interpolation of Wachspress error estimates. *Computers & Mathematics with Applications*, 5(4) :329–336.
- [5] Apprato, D., Arcangeli, R., and Gout, J. (1979b). Sur les éléments finis rationnels de Wachspress. *Numerische Mathematik*, 32(3) :247–270.
- [6] Arbogast, T. and Wang, C. (2022). Direct serendipity and mixed finite elements on convex polygons. *arXiv preprint arXiv :2202.11229*.
- [7] Babuska, I., Strouboulis, T., Babuška, I., Whiteman, J. R., et al. (2001). *The finite element method and its reliability*. Oxford university press.
- [8] Babuška, I. and Suri, M. (1992). On locking and robustness in the finite element method. *SIAM Journal on Numerical Analysis*, 29(5) :1261–1293.
- [9] Baudron, A.-M. and Lautard, J.-J. (2011). Simplified P_n transport core calculations in the Apollo3 system. *Proceedings of International Conference on Mathematics, Computational Methods & Reactor Physics, M&C*.
- [10] Beentjes, C. H. (2015). Quadrature on a spherical surface. *Working note available on the website <http://people.maths.ox.ac.uk/beentjes/Essays>*.
- [11] Björck, Å. (1994). Numerics of gram-schmidt orthogonalization. *Linear Algebra and Its Applications*, 197 :297–316.
- [12] Bourhrara, L. (2019). A new numerical method for solving the Boltzmann transport equation using the P_n method and the discontinuous finite elements on unstructured and curved meshes. *Journal of Computational Physics*, 397 :108801.
- [13] Brezis, H. (2011). *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer.

- [14] Brezzi, F., Cockburn, B., Marini, L. D., and Süli, E. (2006). Stabilization mechanisms in discontinuous Galerkin finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 195(25-28) :3293–3310.
- [15] Brezzi, F., Marini, L. D., and Süli, E. (2004). Discontinuous Galerkin methods for first-order hyperbolic problems. *Mathematical models and methods in applied sciences*, 14(12) :1893–1903.
- [16] Calloo, A., Le Tellier, R., and Labeurthre, D. (2021). High-order finite elements for the neutron transport equation on honeycomb meshes. In *EPJ Web of Conferences*, volume 247, page 03002. EDP Sciences.
- [17] Calloo, A., Vidal, J. F., Le Tellier, R., and Rimpault, G. (2011). Angular finite volume method for solving the multigroup transport equation with piecewise average scattering cross sections. *Proceedings of International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*.
- [18] Carlson, B. G. (1953). Solution of the transport equation by sn approximations. Technical report, Los Alamos Scientific Laboratory.
- [19] Chadwick, J. (1932). The existence of a neutron. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 136(830) :692–708.
- [20] Chandrasekhar, S. (1950). *Radiative Transfer*. Oxford.
- [21] Ciarlet, P. G. (2002). *The finite element method for elliptic problems*, volume 40. Siam.
- [22] Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (2000). The development of discontinuous Galerkin methods. In *Discontinuous Galerkin Methods*, pages 3–50. Springer.
- [23] Collins, B., Stimpson, S., Kelley, B. W., Young, M. T., Kochunas, B., Graham, A., Larsen, E. W., Downar, T., and Godfrey, A. (2016). Stability and accuracy of 3d neutron transport simulations using the 2d/1d method in impact. *Journal of Computational Physics*, 326 :612–628.
- [24] Coste-Delclaux, M. (2006). *Modélisation du phénomène d'autoprotection dans le code de transport multigroupe APOLLO2*. PhD thesis, Paris, CNAM.
- [25] Coste-Delclaux, M., Diop, C., Nicolas, A., and Bonin, B. (2013). *Neutronique*. CEA Saclay ; Groupe Moniteur.
- [26] Cranberg, L., Frye, G., Nereson, N., and Rosen, L. (1956). Fission neutron spectrum of u 235. *Physical Review*, 103(3) :662.
- [27] Cuyt, A., Yaman, I., Ibrahimoglu, B. A., and Benouahmane, B. (2012). Radial orthogonality and lebesgue constants on the disk. *Numerical Algorithms*, 61(2) :291–313.
- [28] Dasgupta, G. and Wachspress, E. L. (2008). The adjoint for an algebraic finite element. *Computers and Mathematics with Applications*, 55 :1988–1997.

- [29] Dautray, R. and Lions, J.-L. (1985). Analyse mathématique et calcul numérique pour les sciences et les techniques. *Collection du Commissariat à l’Energie Atomique. Serie Scientifique*.
- [30] Davidson, G. G. and Palmer, T. S. (2008). Finite element transport using wachspress rational basis functions on quadrilaterals in diffusive regions. *Nuclear science and engineering*, 159 :242–255.
- [31] Di Pietro, D. A., Droniou, J., and Ern, A. (2015). A discontinuous-skeletal method for advection-diffusion-reaction on general meshes. *SIAM Journal on Numerical Analysis*, 53(5) :2135–2157.
- [32] Di Pietro, D. A. and Ern, A. (2011). *Mathematical aspects of discontinuous Galerkin methods*, volume 69. Springer Science & Business Media.
- [33] Duo, J. I. (2008). *Error estimates for nodal and short characteristics spatial approximations of two-dimensional discrete ordinates method*. The Pennsylvania State University.
- [34] Ern, A. and Guermond, J.-L. (2006a). Discontinuous Galerkin methods for Friedrichs’ systems. i. general theory. *SIAM journal on numerical analysis*, 44(2) :753–778.
- [35] Ern, A. and Guermond, J.-L. (2006b). Discontinuous Galerkin methods for Friedrichs’ systems. part ii. second-order elliptic pdes. *SIAM journal on numerical analysis*, 44(6) :2363–2388.
- [36] Ern, A. and Guermond, J.-L. (2008). Discontinuous Galerkin methods for Friedrichs’ systems. part iii. multifield theories with partial coercivity. *SIAM journal on numerical analysis*, 46(2) :776–804.
- [37] Floater, M. S. (2003). Mean value coordinates. *Computer aided geometric design*, 20(1) :19–27.
- [38] Floater, M. S. (2015). Generalized barycentric coordinates and applications. *Acta Numerica*, 24 :161–214.
- [39] Floater, M. S. and Gillette, A. (2017). Nodal bases for the serendipity family of finite elements. *Foundations of Computational Mathematics*, 17(4) :879–893.
- [40] Fortin, A. and Garon, A. (2011). Les éléments finis : de la théorie à la pratique. *Université Laval*.
- [41] Fournier, D. (2011). *Analyse et développement de méthodes de raffinement hp en espace pour l’équation de transport des neutrons*. PhD thesis, Aix-Marseille 1.
- [42] Fournier, D., Herbin, R., and Tellier, R. L. (2013). Discontinuous Galerkin discretization and hp-refinement for the resolution of the neutron transport equation. *SIAM J. Sci. Comput.*, 35 :A936–A956.
- [43] Gastaldo, L., Le Tellier, R., Suteau, C., Fournier, D., and Ruggieri, J. M. (2009). High-order discrete ordinate transport in non-conforming 2D Cartesian meshes. *Proceedings of International Conference on Mathematics, Computational Methods & Reactor Physics*.

- [44] Gear, C. W. (1994). Feature grouping in moving objects. In *Proceedings of 1994 IEEE Workshop on Motion of Non-rigid and Articulated Objects*, pages 214–219. IEEE.
- [45] Giraldo, F. X. (2020). *An Introduction to Element-Based Galerkin Methods on Tensor-Product Bases : Analysis, Algorithms, and Applications*, volume 24. Springer Nature.
- [46] Gout, J. L. (1980). *Éléments finis polygonaux de Wachspress*. PhD thesis, Université de Pau.
- [47] Gout, J. L. (1985). Rational wachspress-type finite elements on regular hexagons. *IMA Journal of Numerical Analysis*, 5 :59–77.
- [48] Guidez, J. and Bonin, B. (2014). *Réacteurs nucléaires à caloporteur sodium*. E-den, Une monographie de la Direction de l'énergie nucléaire. CEA Saclay ; Groupe Moniteur.
- [49] Guyot, M. (2014). Neutronics and thermal-hydraulics coupling : some contributions toward an improved methodology to simulate the initiating phase of a severe accident in a sodium fast reactor. Technical report, Université d'Aix-Marseille.
- [50] Guyot, M., Gubernatis, P., and Tellier, R. L. (2015). Space-time effects in the initiating phase of sodium fast reactors and their evaluation using a three-dimensional neutron kinetics model. *Annals of Nuclear Energy*, 85 :115–126.
- [51] Hackemack, M. W. and Ragusa, J. C. (2018). Quadratic serendipity discontinuous finite element discretization for S_N transport on arbitrary polygonal grids. *Journal of Computational Physics*, 374 :188–212.
- [52] Hébert, A. (2006). High order diamond differencing schemes. *Annals of Nuclear Energy*, 33(17-18) :1479–1488.
- [53] Hébert, A. (2009). *Applied Reactor Physics*. Presses Internationales Polytechnique.
- [54] Hébert, A. (2010). A Raviart-Thomas-Schneider implementation of the simplified P_n method in 3-D hexagonal geometry. In *International Conference on the Physics of Reactor, PHYSOR. American Nuclear Society, Pittsburgh, Pennsylvania, USA*.
- [55] Hernandez, O. M. (2014). *Some contributions towards the parallel simulation of time dependent neutron transport and the integration of observed data in real time*. PhD thesis, Université Pierre et Marie Curie-Paris VI.
- [56] Hormann, K. and Sukumar, N. (2008). Maximum entropy coordinates for arbitrary polytopes. In *Computer Graphics Forum*, volume 27, pages 1513–1520. Wiley Online Library.
- [57] Jaśkowiec, J. (2018). Very high order discontinuous Galerkin method in elliptic problems. *Computational Mechanics*, 62(1) :1–21.
- [58] Jaśkowiec, J., Pluciński, P., and Stankiewicz, A. (2016). Discontinuous Galerkin method with arbitrary polygonal finite elements. *Finite Elements in Analysis and Design*, 120 :1–17.

- [59] Johnson, C. and Pitkäranta, J. (1986). An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. *Mathematics of computation*, 46(173) :1–26.
- [60] Johnson, C. and Pitkäranta, J. (1983). Convergence of a fully discrete scheme for two-dimensional neutron transport. *SIAM journal on numerical analysis*, 20 :951–966.
- [61] Labeurthre, D., Calloo, A., and Le Tellier, R. (2021). Extending Gout’s Wachspress finite elements on regular hexagons to higher orders. In *Mathematics & Computation (M&C) 2021*.
- [62] Labeurthre, D., Calloo, A., and Le Tellier, R. (2022). High-order Wachspress functions on convex polygons through computer algebra. *Journal of Computational Physics*, 470 :111545.
- [63] Lautard, J.-J. and Moller, J.-Y. (2011). Minaret, a deterministic neutron transport solver for nuclear core calculations. *Proceedings of International Conference on Mathematics, Computational Methods & Reactor Physics, M&C*.
- [64] Le Tellier, R. (2019). *Interfaces en physique des réacteurs nucléaires*. PhD thesis, Université Grenoble Alpes.
- [65] Le Tellier, R., Suteau, C., Fournier, D., and Ruggieri, J. (2010). High-order discrete ordinate transport in hexagonal geometry : a new capability in ERANOS. *Il nuovo cimento C*, 33(1) :121–128.
- [66] Leborgne, G. (2007). Complément : espaces de Sobolev fractionnaires.
- [67] Lesaint, P. (1973). Finite element methods for symmetric hyperbolic equations. *Numerische Mathematik*, 21(3) :244–255.
- [68] Lesaint, P. and Raviart, P.-A. (1974). On a finite element method for solving the neutron transport equation. *Publications mathématiques et informatique de Rennes*, pages 1–40.
- [69] Lewis, E. E. (2009). Toward whole-core neutron transport without spatial homogenization. In *Mathematics & Computation (M&C) 2009*.
- [70] Lewis, E. E. and Miller, W. F. (1984). *Computational methods of neutron transport*. John Wiley and Sons, Inc., New York, NY.
- [71] Lions, J. L. and Magenes, E. (1968). *Problèmes aux limites non homogènes et applications*. Dunod.
- [72] Macfarlane, R. E. and Muir, D. W. and George, D. C. (2000). *NJOY99.0 : Code System for Producing Pointwise and Multigroup Neutron and Photon Cross Section from ENDF/B Data*. PSR-480/NJOY99.0, RSICC Peripheral Shielding Routine Collection, Contributed by Los Alamos National Laboratory.
- [73] Massone, M., Abrate, N., Nallo, G. F., Valerio, D., Dulla, S., and Ravetto, P. (2022). Code-to-code SIMMER/FRENETIC comparison for the neutronic simulation of lead-cooled fast reactors. *Annals of Nuclear Energy*, 174 :109124.
- [74] Meurer, A., Smith, C. P., and al. (2017). Sympy : symbolic computing in python. *PeerJ Computer Science*, 3 :e103.

- [75] Mika, J. R. (1961). Neutron transport with anisotropic scattering. *Nuclear Science & Engineering*, 11 :415–427.
- [76] Mirzaee, H., Ji, L., Ryan, J., and Kirby, R. (2011). Smoothness-Increasing Accuracy-Conserving (SIAC) Postprocessing for Discontinuous Galerkin Solutions over Structured Triangular Meshes. *SIAM J. Numerical Analysis*, 49 :1899–1920.
- [77] Mousavi, S. E. and Sukumar, N. (2011). Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. *Computational Mechanics*, 47 :535–554.
- [78] Moustafa, S. (2015). *Massively parallel cartesian discrete ordinates method for neutron transport simulation*. PhD thesis, Bordeaux.
- [79] Mukherjee, T. (2015). *Hierarchical polygonal finite elements of arbitrary order*. PhD thesis, McGill University.
- [80] Pietro, D. A. D., Ern, A., and Lemaire, S. (2016). A review of hybrid high-order methods : formulations, computational aspects, comparison with other methods. *Building bridges : connections and challenges in modern approaches to numerical partial differential equations*, pages 205–236.
- [81] Powar, P. L., Tiwari, R., and Mishra, V. N. (2021). Extension of Dasgupta’s Technique for Higher Degree Approximation. *Universitas Scientiarum*, 26 :139–157.
- [82] Ragusa, J. C., Guermond, J.-L., and Kanschat, G. (2012). A robust S_N -DG-approximation for radiation transport in optically thick and diffusive regimes. *Journal of Computational Physics*, 231(4) :1947–1962.
- [83] Rand, A., Gillette, A., and Bajaj, C. (2014). Quadratic serendipity finite elements on polygons using generalized barycentric coordinates. *Mathematics of computation*, 83 :2691–2716.
- [84] Rank, E., Düster, A., Nübel, V., Preusch, K., and Bruhns, O. T. (2005). High order finite elements for shells. *Computer Methods in Applied Mechanics and Engineering*, 194 :2494–2512.
- [85] Rathod, H. T., Nagaraja, K. V., and Venkatesudu, B. (2007). On the application of two symmetric Gauss-Legendre quadrature rules for composite numerical integration over a triangular surface. *Journal of Applied Mathematics and Computation*, 190 :21–39.
- [86] Reed, W. H. and Hill, T. R. (1973). Triangular mesh methods for the neutron transport equation. Technical report, Los Alamos Scientific Lab., N. Mex.(USA).
- [87] Remacle, J.-F., Flaherty, J. E., and Shephard, M. S. (2003). An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems. *SIAM review*, 45(1) :53–72.
- [88] Reuss, P. (2012). *Précis de neutronique*. EDP sciences.
- [89] Richter, G. R. (1988). An optimal-order error estimate for the discontinuous Galerkin method. *Mathematics of Computation*, 50 :75–88.

- [90] Roache, P. J. (2002). Code verification by the method of manufactured solutions. *J. Fluids Eng.*, 124 :4–10.
- [91] Schneider, D., Dolci, F., Gabriel, F., Palau, J., Guillo, M., Pothet, B., Archier, P., Ammar, K., Auffret, F., Baron, R., et al. (2016). APOLLO3[®] : CEA/DEN deterministic multi-purpose code for reactor physics analysis. In *Proc. PHYSOR*, pages 1–5.
- [92] Sehgal, B. R. et al. (2011). *Nuclear safety in light water reactors : severe accident phenomenology*. Academic Press.
- [93] Sfakianakis, S. (2017). Compute the reduced row echelon form (rref) in python. <https://gist.github.com/sgsfak>.
- [94] Shephard, M. S., Dey, S., and Flaherty, J. E. (1997). A straightforward structure to construct shape functions for variable p -order meshes. *Computer Methods in Applied Mechanics and Engineering*, 147(3-4) :209–233.
- [95] Sibson, R. (1980). A vector identity for the Dirichlet tessellation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 87 :151–155.
- [96] Sovinec, C. R., Glasser, A. H., Gianakon, T. A., Barnes, D. C., Nebel, R. A., Kruger, S. E., Schnack, D. D., Plimpton, S. J., Tarditi, A., and Chu, M. S. (2004). Nonlinear magnetohydrodynamics simulation using high-order finite elements. *Journal of Computational Physics*, 195 :355–386.
- [97] Strang, G. (1993). *Introduction to linear algebra*, volume 3. Wellesley-Cambridge Press Wellesley, MA.
- [98] Sukumar, N. (2004). Construction of polygonal interpolants : a maximum entropy approach. *International journal for numerical methods in engineering*, 61 :2159–2181.
- [99] Szabó, B. and Babuška, I. (2021). *Finite Element Analysis : Method, Verification and Validation*. John Wiley & Sons.
- [100] Takeda, T. and Ikeda, H. (1991). 3-d neutron transport benchmarks. *Journal of Nuclear Science and Technology*, 28 :656–669.
- [101] Tobita, Y., Kondo, S., Yamano, H., Morita, K., Maschek, W., Coste, P., and Cadiou, T. (2006). The development of SIMMER-III, an advanced computer program for LMFR safety analysis, and its application to sodium experiments. *Nuclear Technology*, 153(3) :245–255.
- [102] Vu-Huu, T., Le-Thanh, C., Nguyen-Xuan, H., and Abdel-Wahab, M. (2019). A high-order mixed polygonal finite element for incompressible Stokes flow analysis. *Computer Methods in Applied Mechanics and Engineering*, 356 :175–198.
- [103] Wachspress, E. L. (1975). *A Rational Finite Element Basis*. Academic Press.
- [104] Walters, W. F. (1987). Use of the chebyshev-legendre quadrature set in discrete-ordinate code. Technical Report LA-UR-87-3621, Los Alamos National Laboratory.

- [105] Wang, Y. and Ragusa, J. C. (2010). Diffusion synthetic acceleration for high-order discontinuous finite element S_N transport schemes and application to locally refined unstructured meshes. *Nuclear science and engineering*, 166 :145–166.
- [106] Warren, J., Schaefer, S., Hirani, A. N., and Desbrun, M. (2007). Barycentric coordinates for convex sets. *Advances in computational mathematics*, 27 :319–338.
- [107] Zhang, D. and Rahnema, F. (2020). COMET neutronics solutions to the prismatic VHTR benchmark problem. *Nuclear Engineering and Design*, 358 :110395.
- [108] Zienkiewicz, O. C. and Morice, P. (1971). *The finite element method in engineering science*, volume 1977. McGraw-hill London.

Annexe A

Preuves sur les éléments finis nodaux polygonaux

1 Preuve que les polynômes ne peuvent être utilisés pour générer des éléments finis de Lagrange

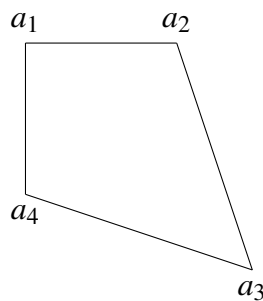


FIGURE A.1 Quadrilatère sans côtés parallèles

Dans le cas d'un polygone à m côtés, $m \geq 4$ ¹, Wachspress dans [103] a prouvé qu'il est impossible de trouver une base polynomiale pour créer un élément finis de Lagrange. Une intuition de la preuve est donnée ci-dessous à l'aide d'un contre exemple dans le cas d'un quadrilatère quelconque *i.e.* sans côtés parallèles.

Pour cela, considérons le quadrilatère avec pour sommets $a_1 = (-1, 1)$, $a_2 = (1, 1)$, $a_3 = (2, -2)$, $a_4 = (-1, -1)$ comme illustré sur la Figure A.1, nous notons v_i les fonctions de

1. Pour des quadrilatères sans côtés parallèles.

base à l'ordre 1. Pour rappel, ces fonctions doivent être linéaires sur leurs côtés adjacents et satisfaire la propriété de Lagrange $v_i(a_j) = \delta_{i,j}$.

Supposons que nous pouvons choisir les fonctions v_i comme dans le cas du parallélogramme pour définir un élément finis de Lagrange. Ces fonctions se définissent alors comme :

$$v_i = c_i \times l_{i+2} l_{i+3} \quad (\text{A.1})$$

où c_i est un coefficient de normalisation tel que $c_i = \frac{1}{l_{i+3}(a_i) \times l_{i+4}(a_i)}$ et l_i est l'équation de la droite passant par les sommets a_{i-1} et a_i .

Considérons la fonction $v_1 = c_1 \times l_3 \times l_4$

Aussi, pour le quadrilatère de la Figure A.1 : $l_1(x,y) = x + 1$, $l_2(x,y) = -y + 1$, $l_3(x,y) = \frac{-3}{4}x - \frac{1}{4}y + 1$, $l_4(x,y) = \frac{1}{2}(x+y) + 1$.

À l'ordre 1, v_1 doit être linéaire sur les arêtes l_1 et l_2 , cependant : $v_1(x, l_2(x,y)) = c_1 \times (\frac{-1}{8}[3x^2 + 4x(-y+1) + (-y+1)^2] + \frac{1}{4}(1-x-y) + 1)$

Ainsi, cette fonction est d'ordre 2. Par conséquent, ajouter un terme polynomial n'aura pas d'ordre impact si ce n'est d'augmenter l'ordre, d'où l'idée d'utiliser des fonctions rationnelles.

2 Preuve de la nécessité d'utiliser des points intérieurs à un ordre $k \geq m$

Soit κ un polygone avec m côtés dont les sommets sont notés $(a_i)_{1 \leq i \leq m}$.

Soit L un polynôme d'ordre m tel que $L(x,y) = \prod_{i=1}^m l_i$ avec l_i l'équation de la droite passant par les points a_{i-1}, a_i . L correspond alors au produit des arêtes constituant le bord du polygone.

Montrons que L ne peut pas se projeter dans la base de Wachspress en l'absence de nœud à l'intérieur de κ . Pour cela, raisonnons par l'absurde.

Considérons alors une base de Wachspress d'ordre m notée W_m , cette dernière consiste en m^2 fonctions, $W_m = \{w_1, \dots, w_m, w_{1_1}, \dots, w_{1_m}, w_{2_1}, \dots, w_{m_m}\}$, chaque fonction étant associée à un nœud tel que $w_i(a_i) = 1$ (respectivement $w_{i_j}(a_{i_j}) = 1$) et 0 sur les autres nœuds. Chaque nœud étant sur le bord de κ .

Or, par définition, $P_m \subset W_m$, par conséquent, le polynôme L devrait pouvoir se projeter exactement dans cette base. Écrivons alors cette projection :

$$L(x, y) = \sum_{i=1}^m (w_i(x, y)L(a_i) + \sum_{j=1}^k w_{ij}L(a_{ij})) \quad (\text{A.2})$$

Cependant, $\forall i, j \in \llbracket 1; m \rrbracket, L(a_i) = L(a_{ij}) = 0 \Leftrightarrow \forall (x, y) \in P, L(x, y) = 0$, ce qui est absurde.

Ainsi, il est impossible de projeter L dans la base des fonctions de Wachspress. En conséquence, il est nécessaire de rajouter un degré de liberté supplémentaire. Cependant, ce nœud ne peut se situer sur les arêtes du polygone. Ainsi, il est nécessaire de rajouter un point intérieur.

3 Propriétés des fonctions de Wachspress sur le bord d'un polygone

Notons $W_k = \{\forall i \in \llbracket 1; m \rrbracket, \forall j \in \llbracket 1; k-1 \rrbracket, w_i = c_i \frac{p^i r^i}{q}; w_{ij} = c_{ij} \frac{p^{ij} r^{ij}}{q}\}$ l'ensemble des fonctions de Wachspress d'ordre k pour un polygone avec m côtés.

Les fonctions w_i, w_{ij} doivent vérifier les contraintes géométriques de l'Equation (2.7). À partir de ces contraintes, si nous notons $\partial\kappa$ le bord du polygone κ , les égalités suivantes peuvent être obtenues :

- $\forall i, l \in \llbracket 1; m \rrbracket, l \notin \{i-1, i, i+1\}, \langle w_i, w_l \rangle_{L^2(\partial\kappa)} = 0$
- $\forall i, l \in \llbracket 1; m \rrbracket, l \notin \{i-1, i\}, \forall j \in \llbracket 1; k-1 \rrbracket, \langle w_i, w_{lj} \rangle_{L^2(\partial\kappa)} = 0$
- $\forall i, l \in \llbracket 1; m \rrbracket, l \neq i, \forall j, p \in \llbracket 1; k-1 \rrbracket, \langle w_{ij}, w_{lp} \rangle_{L^2(\partial\kappa)} = 0$

Annexe B

Définitions des erreurs associées aux propriétés que doivent vérifier les bases

1 Erreur sur les contraintes géométriques

Soient : $\forall i \in \llbracket 1; 6 \rrbracket, \forall j \in \llbracket 1; k-1 \rrbracket, V_i \in \mathbb{R}^{\frac{m \times k(k+1)}{2}}$ et $V_{i_j} \in \mathbb{R}^{\frac{m \times k(k-1)}{2}}$.

Pour un côté i donné, $v_i = (1 - w_i(a_i), \{w_i(a_{l_j}), \forall l \in \{i, i-1\}\})$ et $V_i = [v_j]_{1 \leq j \leq m}$.

Également, pour un côté i donné, $\forall j \in \llbracket 1; k-1 \rrbracket, v_{i_j} = (1 - w_{i_j}(a_{i_j}), \{w_{i_j}(a_{l_l}), \forall l \neq j\})$ et $V_{i_j} = [v_{i_l}]_{1 \leq l \leq m}$.

Notons $\varepsilon_{w_i} = \|V_i\|_2, \varepsilon_{w_{i_j}} = \|V_{i_j}\|_2$, ils représentent respectivement les erreurs sur les contraintes géométriques pour chaque fonction w_i, w_{i_j} .

2 Erreur de projection

Notons :

$$\varepsilon_{\gamma, \beta} = \|x^\gamma y^\beta - \Pi_w x^\gamma y^\beta\|_{L^2(\kappa)} = \sqrt{\int_{\kappa} \left(x^\gamma y^\beta - \sum_{i=1}^m (w_i(x, y) x_i^\gamma y_i^\beta) + \sum_{j=1}^{k-1} w_{i_j}(x, y) x_{i_j}^\gamma y_{i_j}^\beta \right)^2}$$

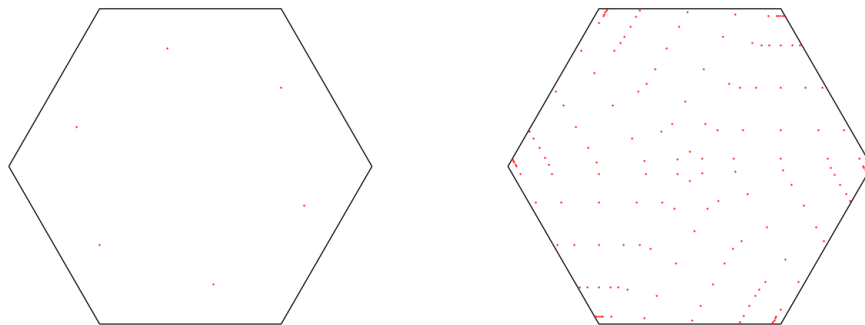
Alors, pour chaque projection, nous introduisons le vecteur suivant $\varepsilon_k = [\varepsilon_{\gamma, \beta}]_{\forall (\gamma, \beta) \in C_k}$.

Finalement, l'erreur totale e_k à un ordre k donné est obtenue comme $e_k = \|\varepsilon_k\|_2$.

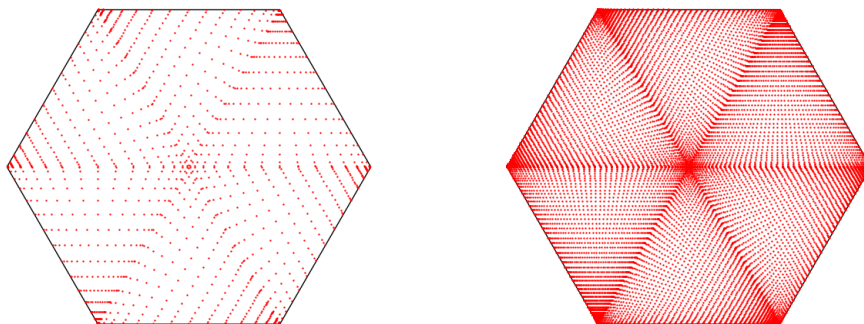
Annexe C

Intégration spatiale sur l'hexagone

Points de quadrature sur un hexagone régulier



(a) Quadratures d'ordres 1 et 5



(b) Quadratures d'ordres 15 et 50

FIGURE C.1 Illustrations des nœuds des quadratures à différents ordres.

Annexe D

Espace V_l ainsi que construction des matrices pour les bases multiplicatives

1 Espace V_l pour la base multiplicative

V_l forme ici une base hybride modale-nodale comme décrit dans [94]. Cet espace se définit comme un ensemble de fonction pour chaque "entité" topologique du maillage *i.e.* des fonctions associées aux nœuds, aux arêtes et aux faces d'un élément. Ces bases ont déjà été appliquées pour la résolution de l'équation de transport [65]. Voici une formalisation de ce type de base :

Soit un polygone κ , soient $M_{\kappa,j}^d$ les ensembles de noeuds ($d = 0$), d'arêtes ($d = 1$), et de faces ($d = 2$); j représente la j -ème entité de l'élément (par exemple, la j -ème arête). Notons les fonctions de bases : $(f_i^{M_{\kappa,j}^d})_i$, ces fonctions de bases doivent vérifier la propriété suivante :

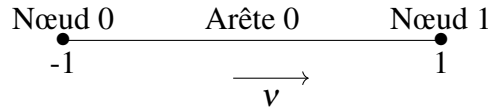
$$\forall \vec{r} \in \{M_{\kappa,j'}^d\}_{j' \neq j} \cup \{M_{\kappa,j'}^{d'}\}_{0 \leq d' < d, j'}, f_i^{M_{\kappa,j}^d}(\vec{r}) = 0 \quad (\text{D.1})$$

Pour vérifier la propriété (D.1), les fonctions doivent se décomposer comme le produit de deux fonctions :

$$f_i^{M_{\kappa,j}^d} = v_{\kappa}^{M_{\kappa,j}^d} g_{i,k}^{M_d} \quad (\text{D.2})$$

- $v_{\kappa}^{M^d, j}$ correspondent aux "blending functions" : fonction indépendante de l'ordre p de la base. Ces fonctions sont associées aux $M_{\kappa, j}^d$ et permettent de vérifier la propriété (D.1).
- $g_{i, k}^{M_d}$ correspondent aux fonctions "mesh entity level", ces fonctions sont définies de manière unique pour M^d indépendamment de la fermeture de l'élément fini auxquels elles appartiennent.

Exemple pour un élément fini de référence 1D :



Une transformation géométrique est réalisée pour définir les fonctions de base sur un élément quelconque $[z_k, z_{k+1}]$ à un élément de référence $[-1, 1]$:

$$\tau^k = \begin{cases} [z_k, z_{k+1}] \iff [-1, 1] \\ z = z_k \frac{1-v}{2} + z_{k+1} \frac{1+v}{2} \iff v = 2 \frac{z - z_k}{z_{k+1} - z_k} - 1 \end{cases}$$

Par conséquent, cet élément fini a 2 "entités" topologiques, une pour les nœuds et une pour l'arête. Ainsi, il nous faut définir 3 fonctions $v_{\kappa}^{M^d, j}$. Par définition, les fonctions $v_{\kappa}^{M^0, j}$ doivent s'annuler sur les nœuds opposés et la fonction $v_{\kappa}^{M^1, 0}$ (vu qu'il n'y a qu'une seule arête) doit s'annuler sur les nœuds appartenant à celle-ci. Ces fonctions sont données au Tableau D.1.

Noeuds	Arêtes
$v_{\kappa}^{M^0, 0} = \frac{1}{2}(1 - v)$	$v_{\kappa}^{M^1, 0} = \frac{1}{4}(1 - v^2)$
$v_{\kappa}^{M^0, 1} = \frac{1}{2}(1 + v)$	x

TABLEAU D.1 Blending functions pour l'élément fini de référence 1D

Les fonctions $g_{i, k}^{M_d}$ sont quant à elles données au Tableau D.2.

Ordre k	Entité	fonction "Mesh entity"
$k = 1$	Nœud	$g_{i, k}^{M^0} = 1$
$k \geq 2$	Arête	$g_{i, k}^{M^1} = \frac{\sqrt{2(2k-1)}}{k(k-1)} \frac{d}{dv} (P_{k-1})(v)$

TABLEAU D.2 fonctions "Mesh entity level" pour l'élément fini de référence 1D

On peut alors écrire V_k sous la forme :

$$V_k = \bigoplus_{m=1}^k \tilde{V}_m \quad (\text{D.3})$$

Où : $\tilde{V}_1 = \{v_{\mathcal{K}}^{M_{k,0}^0}, v_{\mathcal{K}}^{M_{k,1}^0}\}$, $\tilde{V}_2 = \{v_{\mathcal{K}}^{M_{k,0}^1} g_{0,2}^{M_1}\}$, \dots , $\tilde{V}_k = \{v_{\mathcal{K}}^{M_{k,0}^1} g_{0,k}^{M_1}\}$

2 Calcul des différentes matrices avec la base multiplicative

Pour la résolution de l'équation de transport par une méthode DG avec un schéma décentré amont, il est nécessaire de définir l'ensemble des faces pour lesquelles le flux est entrant ou sortant. Il faut introduire de nouvelles notations pour les faces du dessous et du dessus. La Figure D.1 illustre les numérotations de ces faces, elles seront utiles pour calculer les matrices de surface avec les différentes bases.

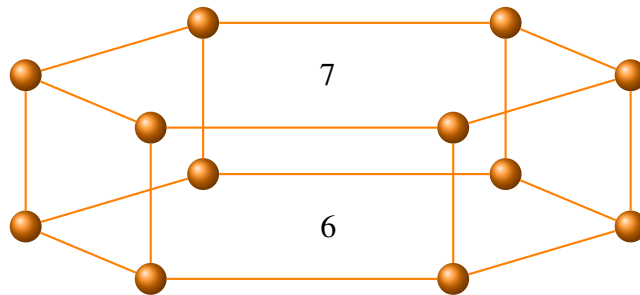


FIGURE D.1 Numérotation des faces du dessous et du dessus pour une maille hexagonale 3D prismatique

2.1 Calcul des matrices de volume

Pour rappel, le vecteur contenant les fonctions de base s'écrit $\underline{v}_{\mathcal{K}}$. Ecrivons alors la matrice de masse en 3D :

$$\begin{aligned}
\underline{\underline{M_K}} &= \int_{\kappa} \underline{v_K}^T \underline{v_K} dV \\
&= \int_{\kappa_{2D}} \int_{\kappa_{1D}} [v_1(x,y)\phi_1(z), \dots, v_N(x,y)\phi_{N_z}(z)]^T [v_1(x,y)\phi_1(z), \dots, v_N(x,y)\phi_{N_z}(z)] dV \\
&= \int_{\kappa_{2D}} \int_{\kappa_{1D}} \begin{bmatrix} v_1^2 \phi_1^2 & (v_1)^2 \phi_1 \phi_2 & \cdots & v_1^2 \phi_1 \phi_{N_z} & v_1 v_2 \phi_1^2 & \cdots & v_1 v_N \phi_1 \phi_{N_z} \\ (v_1)^2 \phi_2 \phi_1 & v_1^2 \phi_2^2 & \cdots & v_1^2 \phi_1 \phi_{N_z} & v_1 v_2 \phi_2 \phi_1 & \cdots & v_1 v_N \phi_2 \phi_{N_z} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ v_N v_1 \phi_{N_z} \phi_1 & v_N v_1 \phi_{N_z} \phi_2 & \cdots & v_N v_1 \phi_{N_z}^2 & \cdots & \cdots & (v_N \phi_{N_z})^2 \end{bmatrix} dV
\end{aligned}$$

En notant respectivement $\underline{\underline{M_{\kappa_{2D}}}}$ et $\underline{\underline{M_{\kappa_{1D}}}}$ les matrices de masse en deux dimensions et en une dimension, la matrice de masse 3D s'écrit comme le produit tensoriel entre la matrice de masse 2D et la matrice de masse 1D.

$$\underline{\underline{M_K}} = \underline{\underline{M_{\kappa_{2D}}}} \otimes \underline{\underline{M_{\kappa_{1D}}}} \quad (D.4)$$

Également, en introduisant respectivement $\underline{\underline{A_{\kappa_{2D}}^i}}$ pour $i \in \{1, 2\}$ et $\underline{\underline{A_{\kappa_{1D}}}}$ les matrices de streaming en deux dimensions et en une dimension, la matrice $\underline{\underline{A_K^i}} = \int_{\kappa} \underline{v_K} \frac{\partial^i \underline{v_K}}{\partial x_i} dV, \forall i \in \llbracket 1, 3 \rrbracket$ est calculée de la manière suivante :

$$\forall i \in \{1, 2\} : \underline{\underline{A_K^i}} = \underline{\underline{A_{\kappa_{2D}}^i}} \otimes \underline{\underline{M_{\kappa_{1D}}}} \quad (D.5)$$

Pour $i = 3$, il faut définir la matrice $\underline{\underline{A_{\kappa_{1D}}}}$:

$$\underline{\underline{A_{\kappa_{1D}}}} = \int_{\kappa_{1D}} \begin{bmatrix} \phi_1 \frac{d\phi_1}{dz} & \phi_1 \frac{d\phi_2}{dz} & \cdots & \phi_1 \frac{d\phi_{N_z}}{dz} \\ \phi_2 \frac{d\phi_1}{dz} & \phi_2 \frac{d\phi_2}{dz} & \cdots & \phi_2 \frac{d\phi_{N_z}}{dz} \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{N_z} \frac{d\phi_1}{dz} & \phi_{N_z} \frac{d\phi_2}{dz} & \cdots & \phi_{N_z} \frac{d\phi_{N_z}}{dz} \end{bmatrix} dV \quad (D.6)$$

Ainsi, on en déduit que :

$$\underline{\underline{A_K^3}} = \underline{\underline{M_{\kappa_{2D}}}} \otimes \underline{\underline{A_{\kappa_{1D}}}} \quad (D.7)$$

2.2 Calcul des matrices de surface

Considérons maintenant les matrices de surface, pour $b \in \partial\kappa_-$, pour cela, commençons par faire un schéma :

On rappelle l'écriture des matrices de surface :

$$\underline{\underline{M}}_{\kappa,b}^{i+} = \int_b n_b^i \underline{v}_\kappa \underline{v}_\kappa^T ds$$

$$\underline{\underline{M}}_{\kappa,b}^{i-} = \int_b n_b^i \underline{v}_\kappa \underline{v}_{\kappa^{b,-}}^T ds$$

b correspond à une face en 3D, si b correspond à l'une des faces 0,1,2,3,4,5, on peut calculer $\underline{\underline{M}}_{\kappa,b}^{i+}$ de la manière suivante :

$$\underline{\underline{M}}_{\kappa,b}^{i+} = \int_{b(y(x),z)} \begin{bmatrix} v_1^2 \phi_1^2 & (v_1)^2 \phi_1 \phi_2 & \cdots & v_1^2 \phi_1 \phi_{N_z} & v_1 v_2 \phi_1^2 & \cdots & v_1 v_N \phi_1 \phi_{N_z} \\ (v_1)^2 \phi_2 \phi_1 & v_1^2 \phi_2^2 & \cdots & v_1^2 \phi_1 \phi_{N_z} & v_1 v_2 \phi_2 \phi_1 & \cdots & v_1 v_N \phi_2 \phi_{N_z} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ v_N v_1 \phi_{N_z} \phi_1 & v_N v_1 \phi_{N_z} \phi_2 & \cdots & v_N v_1 \phi_{N_z}^2 & \cdots & \cdots & (v_N \phi_{N_z})^2 \end{bmatrix} ds \quad (D.8)$$

Cela revient finalement à écrire :

$$\underline{\underline{M}}_{\kappa_{2D},b}^{i+} = \int_{b(y(x))} \begin{bmatrix} (v_1)^2 & v_1 v_2 & \cdots & v_1 v_N \\ v_1 v_2 & (v_2)^2 & \cdots & v_2 v_N \\ \vdots & \vdots & \vdots & \vdots \\ v_N v_1 & v_N v_2 & \cdots & (v_N)^2 \end{bmatrix} ds$$

Et en utilisant la matrice de masse 1D : $\underline{\underline{M}}_{\kappa_{1D}}$, on obtient :

$$\underline{\underline{M}}_{\kappa_{3D},b}^{i+} = \underline{\underline{M}}_{\kappa_{2D},b}^{i+} \otimes \underline{\underline{M}}_{\kappa_{1D}} \quad (D.9)$$

De même :

$$\underline{\underline{M}}_{\kappa_{3D},b}^{i-} = \underline{\underline{M}}_{\kappa_{2D},b}^{i-} \otimes \underline{\underline{M}}_{\kappa_{1D}} \quad (D.10)$$

Le calcul des matrices de surface est différent pour les faces 6 et 7.

Cas où le flux est entrant dans la face 7

Considérons la matrice $\underline{\underline{M}}_{\kappa,b}^{i+}$ pour ces faces dans un premier temps. Commençons par écrire le cas où la cellule est entrante pour la face 7, alors on a :

$$\underline{\underline{M}}_{\kappa,b}^{i+} = \int_{\kappa_{2D}(x,y)} \begin{bmatrix} v_1^2 \phi_1^2 & (v_1)^2 \phi_1 \phi_2 & \cdots & v_1^2 \phi_1 \phi_{N_z} & v_1 v_2 \phi_1^2 & \cdots & v_1 v_N \phi_1 \phi_{N_z} \\ (v_1)^2 \phi_2 \phi_1 & v_1^2 \phi_2^2 & \cdots & v_1^2 \phi_1 \phi_{N_z} & v_1 v_2 \phi_2 \phi_1 & \cdots & v_1 v_N \phi_2 \phi_{N_z} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ v_N v_1 \phi_{N_z} \phi_1 & v_N v_1 \phi_{N_z} \phi_2 & \cdots & v_N v_1 \phi_{N_z}^2 & \cdots & \cdots & (v_N \phi_{N_z})^2 \end{bmatrix} ds \quad (\text{D.11})$$

Toutes les fonctions ϕ_i sont évaluées au point 1.

Ainsi, en faisant apparaître la matrice :

$$\underline{\underline{M}}_{\kappa_{1D},b}^{i+} = \begin{bmatrix} \phi_1^2(1) & \phi_1(1)\phi_2(1) & \cdots & \phi_1(1)\phi_{N_z}(1) \\ \phi_2(1)\phi_1(1) & \phi_2^2(1) & \cdots & \phi_2(1)\phi_{N_z}(1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{N_z}(1)\phi_1(1) & \phi_{N_z}(1)\phi_2(1) & \cdots & \phi_{N_z}^2(1) \end{bmatrix} \quad (\text{D.12})$$

On peut en déduire la matrice $\underline{\underline{M}}_{\kappa,b}^{i+}$:

$$\underline{\underline{M}}_{\kappa,b}^{i+} = \underline{\underline{M}}_{\kappa_{2D}} \otimes \underline{\underline{M}}_{\kappa_{1D},b}^{i+} \quad (\text{D.13})$$

Calculons $\underline{\underline{M}}_{\kappa,b}^{i-}$, vu que le flux va de la face 6 d'une cellule vers la face 7 d'une autre cellule, on introduit la notation ϕ_j^-, v_j^- pour désigner les termes amonts. Par conséquent, on a le produit $\phi_i v_j^-$ qui vaut :

$$\phi_i v_j^- = \begin{cases} 1 & \text{si } i = N_z, j = 1 \\ 0 & \text{sinon} \end{cases}$$

Ecrivons les matrices :

$$\underline{\underline{M_{\kappa 1D,b}^{i-}}} = \begin{bmatrix} \phi_1(1)\phi_1^-(-1) & \phi_1(1)\phi_2^-(-1) & \cdots & \phi_1(1)\phi_{N_z}^-(-1) \\ \phi_2(1)\phi_1^-(-1) & \phi_2(1)\phi_2^-(-1) & \cdots & \phi_2(1)\phi_{N_z}^-(-1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{N_z}(1)\phi_1^-(-1) & \phi_{N_z}(1)\phi_2^-(-1) & \cdots & \phi_{N_z}(1)\phi_{N_z}^-(-1) \end{bmatrix} \quad (\text{D.14})$$

$$\underline{\underline{M_{\kappa 2D}^{i-}}} = \int_{\kappa 2D(x,y)} \begin{bmatrix} v_1 v_1^- & v_1 v_2^- & \cdots & v_1 v_{N_z}^- \\ v_2 v_1^- & v_2 v_2^- & \cdots & v_2 v_{N_z}^- \\ \vdots & \vdots & \vdots & \vdots \\ v_{N_z} v_1^- & v_{N_z} v_2^- & \cdots & v_{N_z} v_{N_z}^- \end{bmatrix} ds \quad (\text{D.15})$$

Or, il est important de remarquer ici que le flux sortant et le flux entrant sont évalués sur le même plan, i.e. $v_i = \phi_i^-$, d'où : $\underline{\underline{M_{\kappa 2D}^{i-}}} = \underline{\underline{M_{\kappa 2D}}}$

On en déduit finalement :

$$\underline{\underline{M_{\kappa,b}^{i-}}} = \underline{\underline{M_{\kappa 2D}^{i-}}} \otimes \underline{\underline{M_{\kappa 1D,b}^{i-}}} = \underline{\underline{M_{\kappa 2D}}} \otimes \underline{\underline{M_{\kappa 1D,b}^{i-}}} \quad (\text{D.16})$$

Cas où le flux est entrant dans la face 6

Dans le cas où le flux est entrant pour la face 6, on a alors $\phi_{N_z} = 1$ et $\phi_i = 0$. On introduit alors la matrice :

$$\underline{\underline{M_{\kappa 1D,b}^{i+}}} = \begin{bmatrix} \phi_1^2(-1) & \phi_1(-1)\phi_2(-1) & \cdots & \phi_1(-1)\phi_{N_z}(-1) \\ \phi_2(-1)\phi_1(-1) & \phi_2^2(-1) & \cdots & \phi_2(-1)\phi_{N_z}(-1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{N_z}(-1)\phi_1(-1) & \phi_{N_z}(-1)\phi_2(-1) & \cdots & \phi_{N_z}^2(-1) \end{bmatrix} \quad (\text{D.17})$$

On peut en déduire la matrice $\underline{\underline{M_{\kappa,b}^{i+}}}$:

$$\underline{\underline{M_{\kappa,b}^{i+}}} = \underline{\underline{M_{\kappa 2D}}} \otimes \underline{\underline{M_{\kappa 1D,b}^{i+}}} \quad (\text{D.18})$$

Pour la matrice $\underline{\underline{M_{\kappa,b}^{i-}}}$, vu que l'on traite maintenant le cas où le flux va de la face 7 d'une cellule vers la face 6 d'une autre cellule, alors on a le produit $\phi_i v_j^-$ qui vaut :

$$\phi_i v_j^- = \begin{cases} 1 & \text{si } i = 1, j = N_z \\ 0 & \text{sinon} \end{cases}$$

On introduit la matrice suivante :

$$\underline{\underline{M_{\kappa 1D,b}^{i-}}} = \begin{bmatrix} \phi_1(-1)\phi_1^-(1) & \phi_1(-1)\phi_2^-(1) & \cdots & \phi_1(-1)\phi_{N_z}^-(1) \\ \phi_2(-1)\phi_1^-(1) & \phi_2(-1)\phi_2^-(1) & \cdots & \phi_2(-1)\phi_{N_z}^-(1) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{N_z}(-1)\phi_1^-(1) & \phi_{N_z}(-1)\phi_2^-(1) & \cdots & \phi_{N_z}(-1)\phi_{N_z}^-(1) \end{bmatrix} \quad (\text{D.19})$$

Puis en considérant comme précédemment la matrice :

$$\underline{\underline{M_{\kappa 2D}^{i-}}} = \int_{\kappa_{2D}(x,y)} \begin{bmatrix} v_1 v_1^- & v_1 v_2^- & \cdots & v_1 v_{N_z}^- \\ v_2 v_1^- & v_2 v_2^- & \cdots & v_2 v_{N_z}^- \\ \vdots & \vdots & \vdots & \vdots \\ v_{N_z} v_1^- & v_{N_z} v_2^- & \cdots & v_{N_z} v_{N_z}^- \end{bmatrix} ds \quad (\text{D.20})$$

On en déduit finalement :

$$\underline{\underline{M_{\kappa,b}^{i-}}} = \underline{\underline{M_{\kappa 2D}^{i-}}} \otimes \underline{\underline{M_{\kappa 1D,b}^{i-}}} = \underline{\underline{M_{\kappa 2D}}} \otimes \underline{\underline{M_{\kappa 1D,b}^{i-}}} \quad (\text{D.21})$$

Annexe E

Résultats Takeda 3D

Taux d'absorption par plan

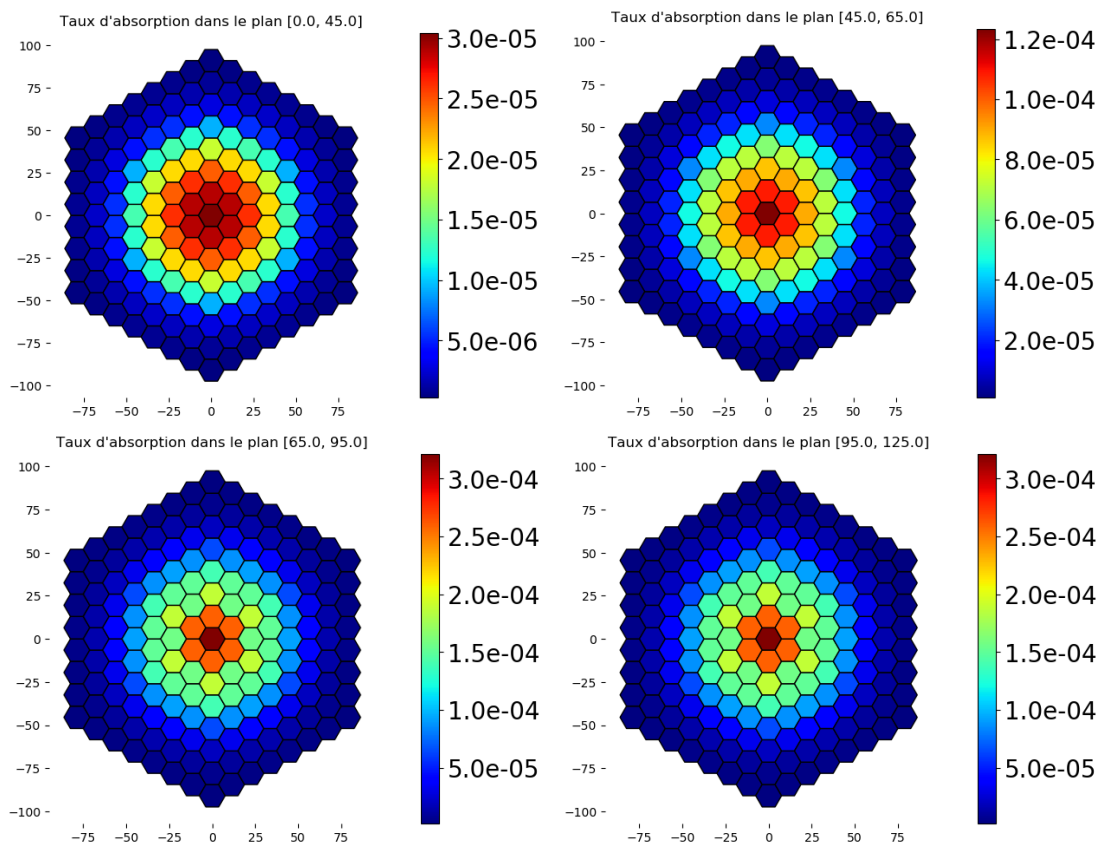


FIGURE E.1 Taux d'absorption dans les quatre premiers plans à partir du bas du cœur.

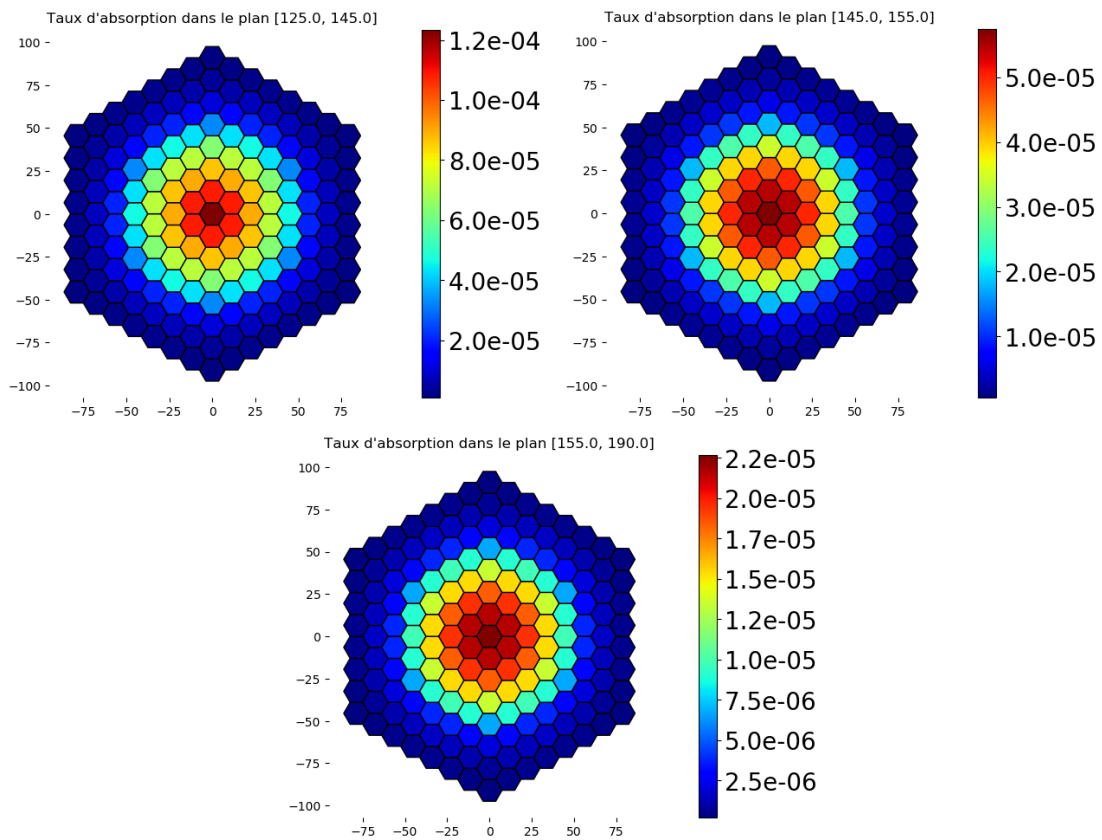


FIGURE E.2 Taux d'absorption dans les trois derniers plans à partir du bas du cœur.