THÈSE DE DOCTORAT DE L'UNIVERSITÉ PARIS XIII

Spécialité : MATHÉMATIQUES APPLIQUÉES

présentée par

Philippe CHARTON

pour obtenir le titre de

Docteur de l'Université de Paris Nord

Sujet :

Produits de matrices rapides en bases d'ondelettes : application à la résolution numérique d'équations aux dérivées partielles

soutenue le 19 janvier 1996

devant le jury composé de :

Rapporteur	Stéphane JAFFARD
Rapporteur	Yvon MADAY
	Mikhaël BALABANE
	Claude BASDEVANT
	Philippe LANGLOIS
	Pierre Gilles LEMARIÉ
	Valérie PERRIER

A Marie-Laure.

Un Grand MERCI à :

- Claude Basdevant, mon directeur de thèse, pour ses compétences, sa grande disponibilité et sa gentillesse ; sans lui cette thèse ne serait pas ce qu'elle est,
- Valérie Perrier, pour son efficacité, sa collaboration et son soutien tout au long de ce travail,
- Mikhaël Balabane, sans qui je n'aurais pu ni commencer ni finir cette thèse,
- Mes deux rapporteurs, Stéphane Jaffard pour ses remarques très bénéfiques et Yvon Maday pour son exigence sur la partie numérique,
- Pierre Gilles Lemarié et Philippe Langlois, pour avoir accepté d'être membre de mon jury,
- Robert Sadourny pour m'avoir accueilli au Laboratoire de Météorologie Dynamique du CNRS,
- Pierre Gigord pour m'avoir accueilli à l'IREMIA, ce qui m'a permis de terminer cette thèse dans de bonnes conditions,
- Armando Babiano pour ses nombreux conseils sur la turbulence et sans qui la page précédente ne serait pas ce qu'elle est,
- Minh Do Khac qui a bien voulu me founir ses programmes de transformée en ondelettes qui ont servi de bases aux miens,
- Eric Goirand pour son programme de calculs de spectres,
- Les autres membres du LMD, et tout particulièrement tous les thésards (Marie, Karine, Patricia, Olivier, François, Thierry et les autres), pour la formidable ambiance qui y règne.

Table des matières

Introduction **7**

1 Propriétés de compression des bases d'ondelettes **11**
1.1 Présentation des bases orthonormées d'ondelettes 11
1.1.1 Analyse multirésolution sur $\mathbb{R}$ 11
1.1.2 Transformée en ondelettes rapide 13
1.1.3 Ondelettes périodiques 15
1.1.4 Analyse multirésolution sur $\mathbb{T}^2$ 16
1.1.5 Interprétation de la Transformée en Ondelettes Discrète (TOD) . . . 21
1.2 Représentation sur bases d'ondelettes de fonctions 22
1.2.1 Théorie 22
1.2.2 Exemples 26
1.3 Représentation sur bases d'ondelettes d'opérateurs 34
1.3.1 Théorie 34
1.3.2 Exemples 35

2 Produits rapides de matrices **41**
2.1 Transformations du système linéaire $Y = BX$ 41
2.1.1 Analyse sur la base de Fourier 41
2.1.2 Analyse en ondelettes – Forme standard 42
2.1.3 Analyse de Beylkin, Coifman et Rokhlin (BCR) – Forme non standard 43
2.2 Implémentation 51
2.2.1 Stockage Morse 51
2.2.2 Produit matrice-vecteur 51
2.2.3 Produit matrice-matrice 52
2.3 Estimation d'erreur 52
2.4 Optimisation dans le cas d'une matrice circulante 53
2.4.1 Décomposition BCR 53
2.4.2 Décomposition en ondelettes 55

2.4.3 Stockage 57
2.4.4 Produit matrice-vecteur 58

3 Applications **61**
3.1 Problèmes modèles 61
3.1.1 Problèmes continus 61
3.1.2 Problèmes discrets 62
3.1.3 Résolution du système linéaire 63
3.2 Expériences numériques en dimension un 63
3.2.1 Présentation des problèmes tests 64
3.2.2 Ajustement des paramètres 67
3.2.3 Comparaison des lacunarités 67
3.2.4 Erreur et temps de calcul en fonctions des seuils sur l'opérateur et la solution 69
3.2.5 Erreur et temps de calcul en fonction de l'ordre de l'ondelette 76
3.2.6 Conclusion 77
3.3 Expériences numériques en dimension deux 78
3.3.1 Factorisation de l'opérateur de la chaleur 78
3.3.2 Schéma en temps 79
3.3.3 Schéma en espace 80
3.3.4 Accélération de la résolution par utilisation des bases d'ondelettes . 80
3.3.5 Accélération du schéma en temps 81
3.3.6 Résultats numériques 83
3.4 Équation de Navier-Stokes bidimensionnelle 88
3.4.1 Présentation des équations 88
3.4.2 Schéma en ondelettes pour Navier-Stokes 89
3.4.3 Résultats numériques 91
3.4.4 Perspectives 93

Conclusion **107**

A **111**
A.1 Calcul de l'inverse d'une matrice circulante 112

Introduction

Les ondelettes sont connues pour être des fonctions qui essayent d'être le meilleur compromis entre la localisation dans l'espace physique et la localisation dans l'espace de Fourier (ce compromis étant limité par le principe d'incertitude d'Heinsenberg). Il existe différentes familles d'ondelettes, ce qui permet de choisir la régularité et la localisation de l'ondelette en fonction de l'utilisation désirée. La transformée en ondelettes continue a été définie par J. Morlet et A. Grossmann dans [Grossmann et al.84]. Peu de temps après, les bases orthonormées d'ondelettes ont vu le jour. Elles sont reliées à la notion d'analyse multirésolution, introduite par S. Mallat et Y. Meyer dans [Mallat89, Meyer90], qui fournit un processus automatique de construction de bases orthonormées d'ondelettes. Mallat construit également dans cet article des algorithmes rapides (comparables à ceux de la transformée de Fourier rapide) pour l'analyse et la synthèse en ondelettes, outil indispensable dans les applications numériques. La construction d'ondelettes à support compact par I. Daubechies dans [Daubechies88] permet d'accélérer encore les algorithmes rapides de transformée en ondelettes discrètes (TOD).

En parallèle, des résultats sur la caractérisation des espaces fonctionnels par l'intermédiaire de la transformée en ondelettes continue ou discrète [Jaffard89, Meyer90, Holschneider et al.91, De Vore et al.92, Donoho93] ont été démontrés. Schématiquement, ils montrent que plus une fonction est régulière localement, plus ses coefficients d'ondelettes en ce point décroissent à petite échelle. Cette propriété conduit à une représentation économique en bases d'ondelettes des champs inhomogènes, présentant des singularités isolées ; les coefficients d'ondelettes de petite échelle étant non nuls uniquement au voisinage des singularités du champ.

Cette propriété permet d'envisager des schémas de résolution d'équations aux dérivées partielles « naturellement adaptatifs » : c'est dans ce contexte (résolution d'EDP par des méthodes d'ondelettes) que se place ce travail.

Plusieurs types d'utilisation des bases orthonormées d'ondelettes pour la résolution des EDP ont vu le jour :

- Le premier type d'utilisation n'utilise pas le concept d'ondelettes mais simplement celui d'analyse multirésolution ; il s'agit de méthodes de Galerkin sur les fonctions d'échelles (fonctions de base des AMR, voir section 1.1.1), ce qui enlève toutes perspectives d'adaptativité [Latto et al.90, Amaratunga et al.93, Dahmen et al.95]. Dans ce cas, les propriétés de convergence des schémas numériques sont du même type que celles obtenues dans le cas des méthodes spectrales, les opérateurs de déri-

vation se discrétisent de la même manière qu'en différences finies.

- On peut également utiliser les ondelettes dans la résolution numérique des EDP comme outil de détection des irrégularités de la solution. Les singularités du champ sont repérées par la décroissance en échelle des coefficients d'ondelettes. Celle-ci est moins rapide aux points des singularités. Ceci peut être appliqué à une méthode adaptative, basée sur une méthode classique, pour détecter les points où la grille doit être raffinée pour simuler correctement la solution [Pathria et al., Jameson94]. Dans ce cas, les ondelettes servent à effectuer un diagnostic (analyse de la solution calculée) plutôt qu'à calculer cette solution.

- Les ondelettes jouent un rôle prépondérant pour l'obtention des matrices d'opérateurs elliptiques de faible conditionnement : dans [Jaffard92] on montre qu'un préconditionneur diagonal permet de rendre le conditionnement de la matrice de l'opérateur $L = (1 - \alpha\Delta)$ borné par une constante indépendante de la dimension du problème quand la matrice est écrite en base d'ondelettes. Ceci aboutit à une résolution efficace du système linéaire associé par une méthode itérative (par exemple la méthode du gradient conjugué). Ce résultat est en fait similaire à celui obtenu par une méthode spectrale ; si nous appliquons la méthode de Fourier à un problème périodique, alors, dans l'espace de Fourier, l'opérateur $(1 - \alpha\Delta)$ est diagonal. Les ondelettes étant un filtre passe-bande, l'opérateur est presque diagonal en base d'ondelettes ; ses valeurs propres sont liées aux valeurs sur la diagonale. Ces propriétés ont été utilisées par [Dahmen et al.92, Dahmen et al.93a, Beylkin93, Lazaar et al.94a].

  Une autre méthode pour résoudre les problèmes elliptiques est de tirer avantage du fait que $L^{-1} = (1 - \alpha\Delta)^{-1}$ est un opérateur de Calderón-Zygmund. La matrice d'un tel opérateur est creuse en ondelettes (section 1.3.1). Deux méthodes peuvent être appliquées pour décomposer l'opérateur en ondelettes : la méthode standard ou la méthode non standard (BCR) [Beylkin et al.91]. Il est clair que ces méthodes de décomposition de l'opérateur L^{-1} ne peuvent être compétitives que sur une série de problèmes elliptiques car elles nécessitent le calcul préalable de la matrice de l'opérateur L^{-1}. Cette méthode peut aussi être appliquée à des équations intégrales comme l'équation d'Helmoltz (voir [Dorobantu94, Rathsfeld94, Kleemann95]).

- On peut aussi construire des méthodes d'ondelettes pour résoudre des problèmes d'évolution. Dans ce cas la solution est décomposée sur des bases d'ondelettes et les équations sont écrites également sur les coefficients d'ondelettes. On sélectionne ensuite les coordonnées en ondelettes significatives de la condition initiale, ce qui permet de réduire notablement le nombre de degrés de liberté utilisés pour le calcul ; la solution est ensuite également représentée par un faible nombre de coefficients d'ondelettes. La plupart des recherches ont été effectuées sur des problèmes de dimension un ; sur l'équation de la chaleur et l'équation de transport pour les problèmes linéaires, sur l'équation de Burgers et des problèmes de combustion pour les problèmes non linéaires.

  Dans [Engquist et al.92, Dorobantu92, Dorobantu et al.95, Dorobantu95], l'équation

de la chaleur unidimensionnelle est résolue. La méthode utilise un schéma explicite en utilisant le fait que les puissances itérées de la matrice représentant l'opérateur $(1+\alpha\Delta)$ sont très creuses pour les puissances élevées si l'on décompose cette matrice par la méthode standard.

Pour la partie hyperbolique d'une EDP, lors de l'utilisation d'un schéma explicite, l'opérateur résolvant est déjà local et creux : les ondelettes n'améliorent pas la représentation des opérateurs. Mais la solution peut présenter des singularités ou des chocs, elle a donc une représentation creuse en ondelettes : c'est en fait la finalité des schémas adaptatifs. Ces méthodes ont été utilisées avec succès dans [Liandrat et al.90b, Maday et al.91, Bacry et al.92, Fröhlich et al.94]. Malgré la simplicité du principe, elles utilisent des stratégies complexes permettant de sélectionner les coefficients significatifs.

Le problème est plus délicat pour les équations non linéaires ou même pour des équations linéaires à coefficients variables car le produit de deux fonctions ne s'exprime pas simplement en terme de coefficients d'ondelettes. Le problème est le même que pour les méthodes spectrales. Comme pour celles-ci, on peut le résoudre de deux manières différentes : par une méthode de collocation ou par une méthode de Galerkin.

Les méthodes de collocation : Dans ces méthodes, pour calculer le produit de deux fonctions, à partir des coefficients d'ondelettes, on revient aux valeurs aux points dans l'espace physique pour faire le produit ; on retourne ensuite dans l'espace des coefficients d'ondelettes. Mais quelle grille choisir dans l'espace physique ? On peut utiliser une grille uniforme correspondant à la plus petite échelle de la base d'ondelettes et utiliser les algorithmes rapides, la TOD, mais on perd l'adaptativité. Pour conserver l'adaptativité, c'est à dire garder seulement un petit nombre de degrés de liberté pour représenter la solution, nous devons définir une grille dans l'espace physique correspondant aux coefficients d'ondelettes sélectionnés ainsi qu'une transformée en ondelettes adaptée (la TOD classique ne peut pas être utilisée dans ce contexte). Des approches de ce problème ont été proposées dans [Ponenti94, Fröhlich et al.94, Lemarié95].

Les méthodes de Galerkin : Les termes non linéaires peuvent également être calculés directement dans l'espace des coefficients d'ondelettes en évaluant les coefficients d'interactions. Par exemple, pour calculer u^2, dans l'espace des coefficients d'ondelettes, nous avons :

$$c_\alpha(u^2) = \sum_{\beta,\gamma} \Gamma^{\beta}_{\alpha\ \gamma} c_\beta(u) c_\gamma(u),$$

les coefficients d'interactions entre les ondelettes ψ_β et les ondelettes ψ_γ dans la base d'ondelettes ψ_α étant donnés par $\Gamma^{\beta}_{\alpha\ \gamma} = \langle \psi_\beta \psi_\gamma | \psi_\alpha \rangle$. La complexité a priori de ce calcul est très élevée, beaucoup plus que la TOD nécessitée par la méthode de collocation. Mais avec des ondelettes à support compact, et en tenant compte du fait que les ondelettes sont des filtres passe-bande, on a seulement un nombre

restreint de coefficients d'interactions à calculer. Néanmoins, bien que sa complexité asymptotique soit inférieure, cette méthode est dans la pratique plus coûteuse que la précédente.

– D'autres approches du problème utilisant des bases d'ondelettes non orthogonales sont également possibles. Par exemple, les méthodes de Petrov-Galerkin, associées à des bases biorthogonales d'ondelettes, consistent à utiliser des bases différentes pour l'analyse et la synthèse de la solution (voir [Liandrat et al.91, Lazaar et al.94b, Lazaar et al.94a, Dahlke et al.94] pour les méthodes numériques et [Cohen et al.92] pour les bases biorthogonales d'ondelettes). La résolution d'un problème elliptique dans une base biorthogonale adaptée se résume alors à un simple changement de base.

L'objectif de ce travail est le développement d'algorithmes permettant d'effectuer **rapidement** des produits matrice-vecteur et matrice-matrice, et d'insérer ensuite ces algorithmes dans la résolution d'équations aux dérivées partielles en dimension un et deux, la finalité étant la résolution des équations de Navier-Stokes. Les méthodes présentées dans ce mémoire de thèse pour résoudre des EDP se servent des ondelettes comme un outil de compression : nous utiliserons des schémas aux différences finies en espace, des schémas implicites en temps pour des raisons de stabilité, et montrerons comment l'utilisation des ondelettes permet de rendre ceux-ci plus rapides grâce à la compression obtenue **à la fois sur l'opérateur et sur la solution**. Nous montrerons que la méthode des puissances itérées s'applique également aux cas des schémas implicites.

Le plan employé est le suivant : dans le premier chapitre, après une brève présentation des analyses multirésolutions qui sont la source des bases d'ondelettes et des algorithmes associés, nous présentons les justifications théoriques des propriétés de compressions des analyses en ondelettes que nous illustrons sur des exemples en comparant avec la base de Fourier. Nous décrivons ensuite, dans le chapitre deux, des méthodes pour calculer efficacement dans la pratique des produits matrice-vecteur et matrice-matrice creux. Enfin, dans le chapitre trois, nous appliquons cette méthode à la résolution d'équations aux dérivées partielles, en ayant d'abord réduit l'équation à de simples calculs de produits matriciels. Cette méthode est appliquée avec succès à la simulation directe des équations de Navier-Stokes bidimensionnelles incompressibles.

Chapitre 1

Propriétés de compression des bases d'ondelettes

Dans cette partie, nous présentons l'analyse en ondelettes : c'est l'outil qui va nous permettre de comprimer l'information afin de réaliser de manière rapide nos produits de matrices. L'analyse en ondelettes s'introduit naturellement à partir de la notion d'analyse multirésolution ; il s'agit d'approcher une fonction F par une suite f_j de fonctions simples, f_j étant caractérisée par ses valeurs sur le réseau $2^{-j}\mathbb{Z}$.

1.1 Présentation des bases orthonormées d'ondelettes

La construction des bases orthonormées d'ondelettes de $L^2(\mathbb{R})$ ($L^2(\mathbb{R})$ désigne ici l'ensemble des fonctions de $\mathbb{R}$ dans $\mathbb{R}$ de carré intégrable) est liée à la notion d'analyse multirésolution [Daubechies88, Meyer90, Daubechies92], c'est à dire à la définition d'une suite de sous-espaces emboîtés $(V_j)_{j\in\mathbb{Z}}$ invariants par translations, se déduisant par changement d'échelle dyadique, et approchant $L^2(\mathbb{R})$. Les ondelettes apparaissent comme des bases des supplémentaires orthogonaux $V_j \backslash V_{j-1}, V_{j-1} \backslash V_{j-2}, \cdots$.

Ces espaces V_j permettent l'approximation de solutions d'EDP par des méthodes de Galerkin. En dimension un, la solution est cherchée sous la forme d'une somme d'ondelettes monodimensionnelles, les opérateurs pouvant être discrétisés sur des bases d'ondelettes bidimensionnelles.

Nous présentons successivement les analyses multirésolutions sur $\mathbb{R}$ et leurs algorithmes, puis les ondelettes périodiques en dimensions un et deux.

1.1.1 Analyse multirésolution sur $\mathbb{R}$

Définition 1 *Une analyse multirésolution (AMR) sur $\mathbb{R}$ est une décomposition de l'espace de Hilbert $L^2(\mathbb{R})$, en une suite croissante de sous-espaces vectoriels fermés V_j vérifiant :*

$$\forall j \in \mathbb{Z}, \quad V_j \subset V_{j+1} \tag{1.1}$$

$$\bigcap_{j\in\mathbb{Z}} V_j = \{0\} \quad et \quad \bigcup_{j\in\mathbb{Z}} V_j \ est\ dense\ dans\ \mathbf{L}^2(\mathbb{R}) \tag{1.2}$$

$$\forall f \in L^2(\mathbb{R})\ \forall j \in \mathbb{Z}, \quad f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1} \tag{1.3}$$

$$\exists \varphi \in V_0\ telle\ que \{\varphi(x-k)\}_{k\in\mathbb{Z}}\ soit\ une\ base\ orthonormée\ de\ V_0. \tag{1.4}$$

Remarques :

- Cette fonction φ sera appelée fonction d'échelle.
- Certains auteurs remplacent la propriété (1.4) par :

$$\exists\ g \in V_0\ telle\ que \{g(x-k)\}_{k\in\mathbb{Z}}\ soit\ une\ base\ de\ Riesz\ de\ V_0. \tag{1.5}$$

  mais on peut montrer que ces deux définitions sont équivalentes φ étant construite aisément à partir de g (voir par exemple [Meyer90] §II.3).
- De la propriété (1.4) (ou (1.5)), on déduit la propriété d'invariance par translation unitaire :

$$\forall f \in L^2(\mathbb{R})\ \forall k \in \mathbb{Z}, f(x) \in V_0 \Leftrightarrow f(x-k) \in V_0,$$

- de (1.3) et de (1.4) que $\{2^{\frac{j}{2}}\varphi(2^j x - k)\}_{k\in\mathbb{Z}}$ est une base orthonormée de V_j.
- Cette AMR est dite r-régulière si on peut choisir φ (ou g) de sorte que l'on ait :

$$\varphi\ est\ dérivable\ jusqu'à\ l'ordre\ r\ et\ ses\ dérivées\ sont\ à\ décroissance\ rapide. \tag{1.6}$$

On définit ensuite W_j comme le supplémentaire orthogonal de V_j dans V_{j+1} :

$$V_{j+1} = V_j \overset{\perp}{\bigoplus} W_j \tag{1.7}$$

Cet espace W_j contient les détails d'échelle $2^{-(j+1)}$ nécessaires pour passer de l'approximation à l'échelle 2^{-j} (dans V_j) à celle à l'échelle $2^{-(j+1)}$ (dans V_{j+1}). On a donc la décomposition de $L^2(\mathbb{R})$ suivante :

$$L^2(\mathbb{R}) = \overset{\perp}{\bigoplus_{j\in\mathbb{Z}}} W_j$$

L'espace W_j vérifie également la propriété d'invariance par échelle :

$$\forall f \in L^2(\mathbb{R}), \forall j \in \mathbb{Z},\ f(x) \in W_j \Leftrightarrow f(2x) \in W_{j+1}$$

A partir de φ on peut définir une fonction ψ telle que $\{\psi(x-k)\}_{k\in\mathbb{Z}}$ soit une base orthonormée de W_0. $\{\psi_k^j(x) = 2^{\frac{j}{2}}\psi(2^j x - k)\}_{j\in\mathbb{Z},k\in\mathbb{Z}}$ est alors une base orthonormée de

$L^2(\mathbb{R})$. La fonction ψ est appelée ondelette, génératrice de la base d'ondelettes (ψ_k^j). Dans le cas d'une analyse multirésolution r-régulière la fonction ψ vérifie également la propriété (1.6) ainsi que la propriété suivante dite de « moments nuls » :

$$\int x^q \psi(x)\, dx = 0 \text{ pour } 0 \leq q \leq r. \tag{1.8}$$

Un exemple trivial d'AMR est donné par la base de Haar ; celle-ci a pour fonction d'échelle la fonction indicatrice de l'intervalle $[0,1]$, $\varphi = 1_{[0,1]}$. L'ondelette associée est $\psi = 1_{[0,1/2]} - 1_{[1/2,1]}$.

Les propriétés (1.6) et (1.8) sont essentielles pour caractériser la décroissance des coefficients d'ondelettes d'une fonction par sa régularité globale et locale, et permettre des représentations « économiques » en série d'ondelettes (voir section 1.2.1.

Dans la pratique, on utilise le plus souvent soit les ondelettes splines qui permettent de calculer les valeurs ponctuelles $u(i\delta x)$ d'une fonction u à partir de ses coefficients sur la base $\varphi(x-i)$ (grâce à l'existence des bases d'interpolation spline), soit des ondelettes à support compact pour lesquelles on dispose d'algorithmes d'analyse en ondelettes de complexités théoriques plus faibles. Les détails de construction de ces bases d'ondelettes sont explicités dans [Meyer90] et [Daubechies92].

1.1.2 Transformée en ondelettes rapide

Les algorithmes d'analyse et de synthèse sur bases orthonormées d'ondelettes sont des algorithmes rapides, basés sur des convolutions avec des filtres discrets, ils sont dûs à [Mallat89]. Le calcul de N coefficients s'effectue en $O(N)$ opérations si les ondelettes sont à support compact, sinon en $O(N \log_2 N)$ opérations (dans ce dernier cas, les convolutions sont effectuées à l'aide de la FFT).

Le calcul de la transformée d'une fonction suivant une base d'ondelettes $\{\psi_k^j, j \in \mathbb{Z}, k \in \mathbb{Z}\}$ utilise la structure multiéchelle des espaces $(V_j)_{j\in\mathbb{Z}}$ d'une analyse multirésolution. De l'équation (1.3) (invariance par dilatation) et des inclusions $V_0 \subset V_1, W_0 \subset V_1$, on déduit l'existence de filtres discrets $h = (h_n)_{n\in\mathbb{Z}}$ et $g = (g_n)_{n\in\mathbb{Z}}$ vérifiant :

$$\varphi(x) = \sqrt{2} \sum_{k\in\mathbb{Z}} h_k\, \varphi(2x-k) \text{ avec } h_k = \int_{-\infty}^{+\infty} \varphi(x) \cdot \sqrt{2}\, \varphi(2x-k)\, dx \tag{1.9}$$

$$\psi(x) = \sqrt{2} \sum_{k\in\mathbb{Z}} g_k\, \varphi(2x-k) \text{ avec } g_k = \int_{-\infty}^{+\infty} \psi(x) \cdot \sqrt{2}\, \varphi(2x-k)\, dx \tag{1.10}$$

Un choix possible pour le filtre g est (le filtre h étant imposé par les espaces V_j) :

$$g_k = (-1)^k h_{1-k} \tag{1.11}$$

h et g sont appelés filtres miroirs en quadrature. Dans le cas des ondelettes à support compact, ils ont une longueur finie L.

Ces filtres vont permettre de calculer récursivement les coefficients d'ondelettes de la fonction f à l'aide du schéma suivant :

Soient s_k^j et d_k^j respectivement les coordonnées de f dans la base $\{\varphi_k^j = 2^{\frac{j}{2}}\varphi(2^j x - k), k \in \mathbb{Z}\}$ de V_j et la base $\{\psi_k^j = 2^{\frac{j}{2}}\psi(2^j x - k), k \in \mathbb{Z}\}$ de W_j. Comme ces bases sont orthonormées, s_k^j et d_k^j sont donnés par les produits scalaires :

$$s_k^j = \int_{-\infty}^{+\infty} f(x)\varphi_k^j(x)dx \quad \text{et} \quad d_k^j = \int_{-\infty}^{+\infty} f(x)\psi_k^j(x)dx \tag{1.12}$$

Les coefficients $(s_k^{j-1})_{k\in\mathbb{Z}}$ et $(d_k^{j-1})_{k\in\mathbb{Z}}$ se déduisent de $(s_k^j)_{k\in\mathbb{Z}}$ par les convolutions discrètes suivies de décimations avec les filtres h et g (convolution habituelle mais on décale l'entrée de deux points à chaque étape) :

$$s_k^{j-1} = \sum_{n\in\mathbb{Z}} h_n s_{2k+n}^j \tag{1.13}$$

$$d_k^{j-1} = \sum_{n\in\mathbb{Z}} g_n s_{2k+n}^j \tag{1.14}$$

Ces formules permettent de calculer les coefficients de f sur les bases $(\psi_k^{j-1})_k$ de W_{j-1} et $(\varphi_k^{j-1})_k$ de V_{j-1} à partir des coefficients de f sur la base $(\varphi_k^j)_k$ de V_j. Connaissant la décomposition $(s_k^0)_{k=0,N-1}$ d'une fonction de l'espace V_0 suivant les $\varphi(x-k)$, on peut calculer N coefficients d'ondelettes dans les espaces $W_j, j \leq 0$ par l'algorithme récursif suivant :

$$\begin{array}{ccccccccc} (s_k^0)_{k=0,N-1} & \longrightarrow & (s_k^{-1})_{k=0,\frac{N}{2}-1} & \longrightarrow & (s_k^{-2})_{k=0,\frac{N}{4}-1} & \longrightarrow & (s_k^{-3})_{k=0,\frac{N}{8}-1} & \cdots \\ & \searrow & & \searrow & & \searrow & & \\ & & (d_k^{-1})_{k=0,\frac{N}{2}-1} & & (d_k^{-2})_{k=0,\frac{N}{4}-1} & & (d_k^{-3})_{k=0,\frac{N}{8}-1} & \cdots \end{array} \tag{1.15}$$

Cette étape sera appelée analyse, l'opération inverse sera appelée synthèse. Dans cet algorithme, on se limite volontairement au calcul de $N-1$ coefficients d'ondelettes $(d_k^{-1})_{k=0,\frac{N}{2}-1} \cdots (d_0^{-\ln_2 N})$ plus une valeur « moyenne » $(s_0^{-\ln_2 N})$, soit en tout N coefficients, étant parti de N valeurs $(s_k^0)_{k=0,N-1}$. On ne calcule pas les coefficients (d_k^j) en dehors du support $[0, N]$.

La formule de synthèse, déduite de (1.7) est la suivante :

$$s_k^{j+1} = \sum_{n\in\mathbb{Z}} h_{k-2n} s_n^j + \sum_{n\in\mathbb{Z}} g_{k-2n} d_n^j \tag{1.16}$$

et l'on utilise l'algorithme pyramidal (1.15) dans le sens inverse. A cette étape, les coefficients d'ondelettes « omis » dans l'analyse entraînent des erreurs lors de la synthèse aux bords de l'intervalle d'étude $[0, N]$.

Pour effectuer l'analyse ou la synthèse d'une fonction à support dans un intervalle, le choix des ondelettes sur $\mathbb{R}$ n'est pas judicieux : on peut par contre utiliser des ondelettes

sur l'intervalle [Meyer92, Cohen et al.93]; ces ondelettes ne sont plus invariantes par translation. Une autre possibilité est de construire des ondelettes périodiques [Perrier et al.89, Meyer90]; ces ondelettes ne sont plus invariantes par changement d'échelle (pour les échelles proches de la période). Dans la suite, nous nous intéresserons au cas périodique.

1.1.3 Ondelettes périodiques

La construction des ondelettes périodiques sur $[0,1]$ (bases orthonormées de $L^2(\mathbb{T})$) est similaire à celle des ondelettes sur $\mathbb{R}$. La différence essentielle réside dans la perte de l'invariance par échelle. Toutefois, asymptotiquement quand l'échelle diminue, on retrouve les ondelettes sur $\mathbb{R}$.

Définition 2 *Une analyse multirésolution r-régulière sur le tore $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ est une décomposition de $L^2(\mathbb{T})$ en une suite croissante de sous-espaces vectoriels fermés V_j vérifiant*

$$\forall j \in \mathbb{N}, \qquad V_j \subset V_{j+1} \tag{1.17}$$

$$\bigcup_{j\in\mathbb{N}} V_j \text{ est dense dans } \mathbf{L}^2(\mathbb{T}) \tag{1.18}$$

$$V_0 = \{f = C^{te}\} \tag{1.19}$$

$$f(x) \in V_j \Rightarrow f(2x) \in V_{j+1} \tag{1.20}$$

$$f(x) \in V_{j+1} \Rightarrow f\left(\frac{x}{2}\right) + f\left(\frac{x}{2}+\frac{1}{2}\right) \in V_j. \tag{1.21}$$

$$\varphi \text{ est dérivable jusqu'à l'ordre } r. \tag{1.22}$$

$dim V_j = 2^j$, et il existe pour tout j une base orthonormée de $V_j, (\varphi_k^j)_{0\le k<2^j}$, de la forme :

$$\varphi_k^j(x) = \varphi_0^j(x - \frac{k}{2^j}) \tag{1.23}$$

On montre que l'on peut construire une telle analyse multirésolution de $L^2(\mathbb{T})$ en périodisant une analyse multirésolution de $L^2(\mathbb{R})$, soit en posant :

$$\varphi_k^j(x) = 2^{\frac{j}{2}} \sum_{l\in\mathbb{Z}} \varphi(2^j(x-l)-k)$$

et

$$\psi_k^j(x) = 2^{\frac{j}{2}} \sum_{l\in\mathbb{Z}} \psi(2^j(x-l)-k)$$

où φ est la fonction d'échelle d'une analyse multirésolution de $L^2(\mathbb{R})$. Pour les calculs de transformées en ondelettes, il suffit d'utiliser les formules sur $\mathbb{R}$ (1.13-1.14) en périodisant

les filtres à chaque échelle 2^j :

$$s_i^{j-1} = \sum_{n=0}^{2^j-1} h_n^j s_{2i+n}^j \qquad \text{avec} \qquad h_n^j = \sum_{k\in\mathbb{Z}} h_{n+2^j k} \tag{1.24}$$

$$d_i^{j-1} = \sum_{n=0}^{2^j-1} g_n^j s_{2i+n}^j \qquad \text{avec} \qquad g_n^j = \sum_{k\in\mathbb{Z}} g_{n+2^j k} \tag{1.25}$$

Ainsi, une fonction de V_p se décompose suivant :

$$V_p = V_0 \bigoplus_{j=0}^{p-1} W_j$$

sa composante s_0^0 sur V_0 étant sa valeur moyenne sur $\mathbb{T}$.

Remarque :

Contrairement au cas sur $\mathbb{R}$, les filtres dépendent de l'échelle j. Le coût de l'algorithme est de $2LN$ opérations dans le cas des ondelettes à support compact (L longueur des filtres h et g, N longueur du vecteur), et de $3N \log_2 N$ pour les ondelettes splines.

1.1.4 Analyse multirésolution sur $\mathbb{T}^2$

Comme pour la dimension un, on définit une analyse multirésolution sur $\mathbb{T}^2$ par une suite $(V_j(\mathbb{T}^2))$ de sous espaces vectoriels fermés de $L^2(\mathbb{T}^2)$ vérifiant des propriétés analogues à (1.17-1.23). Une telle analyse multirésolution peut être obtenue par simple produit tensoriel d'analyses unidimensionnelles V_j i.e. : $V_j(\mathbb{T}^2) = V_j \otimes V_j$. On obtient alors :

$$V_{j+1}\otimes V_{j+1} = (V_j \oplus W_j)\otimes(V_j \oplus W_j) = (W_j \otimes W_j)\oplus(W_j \otimes V_j)\oplus(V_j \otimes W_j)\oplus(V_j \otimes V_j) \tag{1.26}$$

Soit $W_i(\mathbb{T}^2)$ le supplémentaire orthogonal de $V_i(\mathbb{T}^2)$ dans $V_{i+1}(\mathbb{T}^2)$, on déduit de l'expression (1.26) :

$$W_j(\mathbb{T}^2) = (W_j \otimes W_j) \oplus (W_j \otimes V_j) \oplus (V_j \otimes W_j) \tag{1.27}$$

La base d'ondelettes de $L^2(\mathbb{T}^2)$ sera donc engendrée par trois ondelettes génératrices différentes, $\psi(x)\psi(y)$, $\psi(x)\varphi(y)$ et $\varphi(x)\psi(y)$, que nous appellerons $\left(\psi^{(i)}(x,y)\right)_{1\le i\le 3}$, engendrant trois sortes d'espaces $\left(W_j^i(\mathbb{T}^2)\right)_{1\le i\le 3}$.

L'algorithme d'analyse d'une fonction de deux variables sur une base d'ondelettes est donc le suivant : si l'on note $\alpha^j, \beta^j, \gamma^j$ et s^j ses coefficients respectivement sur $(W_j \otimes W_j), (W_j \otimes V_j), (V_j \otimes W_j)$ et $(V_j \otimes V_j)$, on obtient les formules permettant de passer de l'espace $V_j(\mathbb{T}^2)$ aux espaces $W_{j-1}^l(\mathbb{T}^2)$ et $V_{j-1}(\mathbb{T}^2)$:

$$0 \le i,l \le 2^{j-1}-1 \qquad \alpha_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} g_k^j g_m^j s_{k+2i,m+2l}^j \tag{1.28}$$

$$0 \leq i,l \leq 2^{j-1}-1 \qquad \beta_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} g_k^j h_m^j s_{k+2i,m+2l}^j \tag{1.29}$$

$$0 \leq i,l \leq 2^{j-1}-1 \qquad \gamma_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} h_k^j g_m^j s_{k+2i,m+2l}^j \tag{1.30}$$

$$0 \leq i,l \leq 2^{j-1}-1 \qquad s_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} h_k^j h_m^j s_{k+2i,m+2l}^j \tag{1.31}$$

où (g_k^j) et (h_k^j) sont les filtres unidimensionnels introduits en 1.1.3 (1.24-1.25). La formule de synthèse est la suivante (reconstruction dans $V_j(\mathbb{T}^2)$ à partir des décompositions dans $W_{j-1}^l(\mathbb{T}^2)$ et $V_{j-1}(\mathbb{T}^2)$) :
$0 \leq s,l \leq 2^j$

$$s_{s,l}^j = \sum_{n,m=0}^{2^{j-1}} \left(h_{s-2n}^j h_{l-2m}^j s_{n,m}^{j-1} + g_{s-2n}^j g_{l-2m}^j \alpha_{n,m}^{j-1} + g_{s-2n}^j h_{l-2m}^j \beta_{n,m}^{j-1} + h_{s-2n}^j g_{l-2m}^j \gamma_{n,m}^{j-1} \right)$$

L'algorithme utilisé a la même structure pyramidale que pour la dimension un. La représentation d'un champ bidimensionnel analysé en ondelettes est schématisé sur la figure 1.1. L'extension aux dimensions supérieures est également possible, dans $\mathbb{R}^n$ on a n^2-1 ondelettes différentes.

Nous présentons (figures 1.2 et suivantes) une analyse multirésolution d'un champ turbulent bidimensionnel.

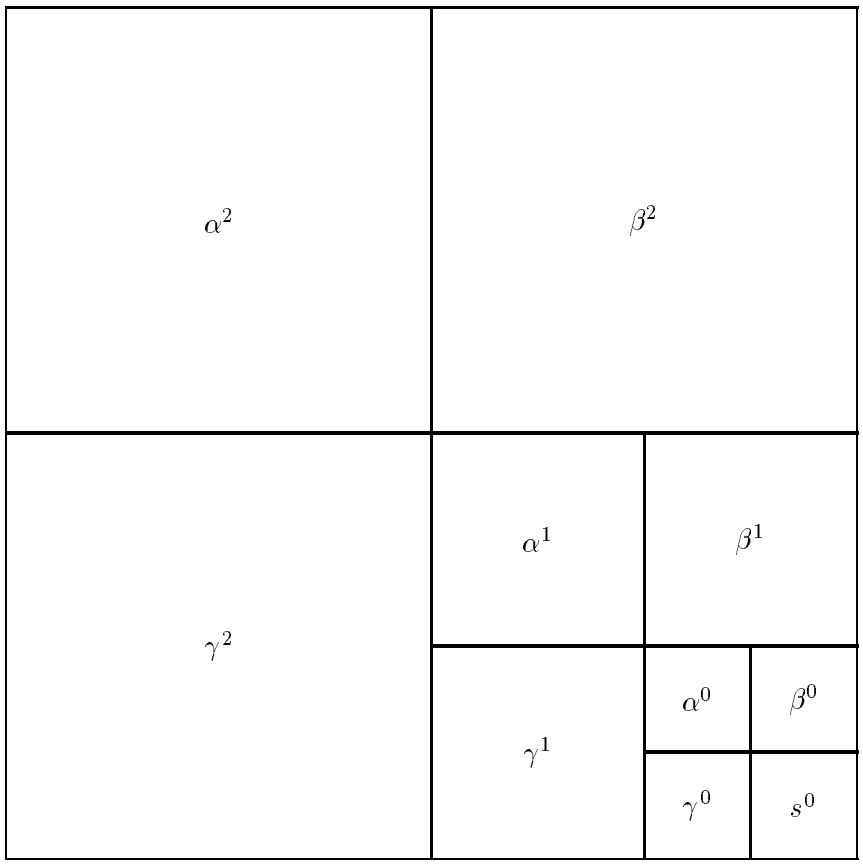


FIG. 1.1 – *Analyse en ondelettes de $L^2(\mathbb{T}^2)$ d'un champ bidimensionnel: la taille des carrés est proportionnelle au nombre de coefficients contenus dans ces carrés.*

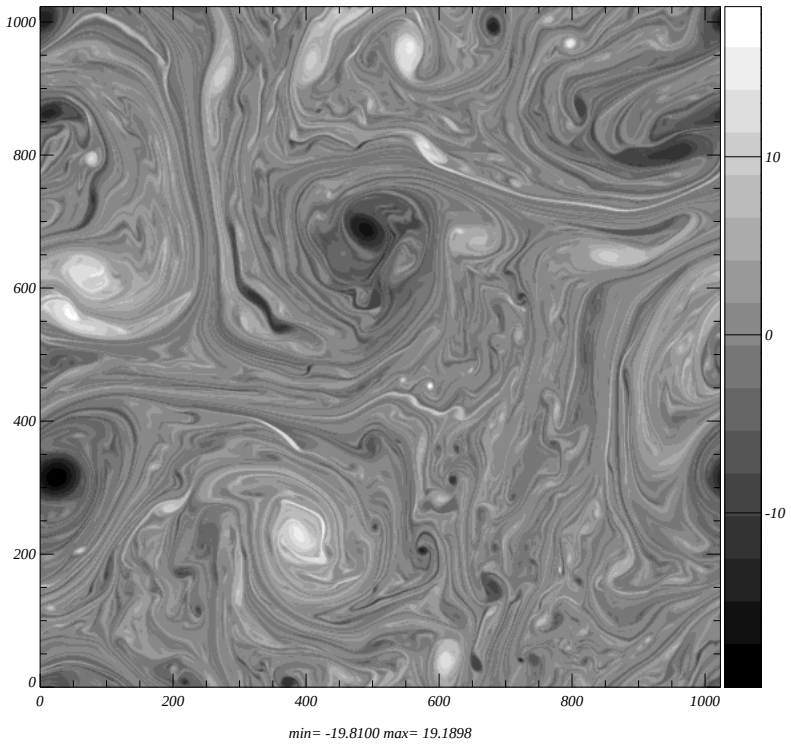


FIG. 1.2 – *Champ projeté dans* V_{10}

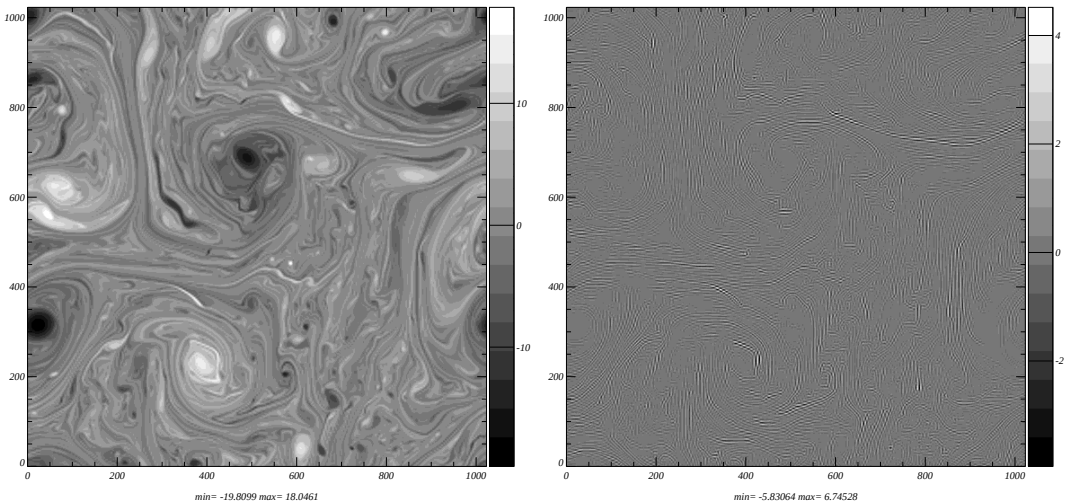


FIG. 1.3 – *champ projeté dans* V_8 *et* W_8

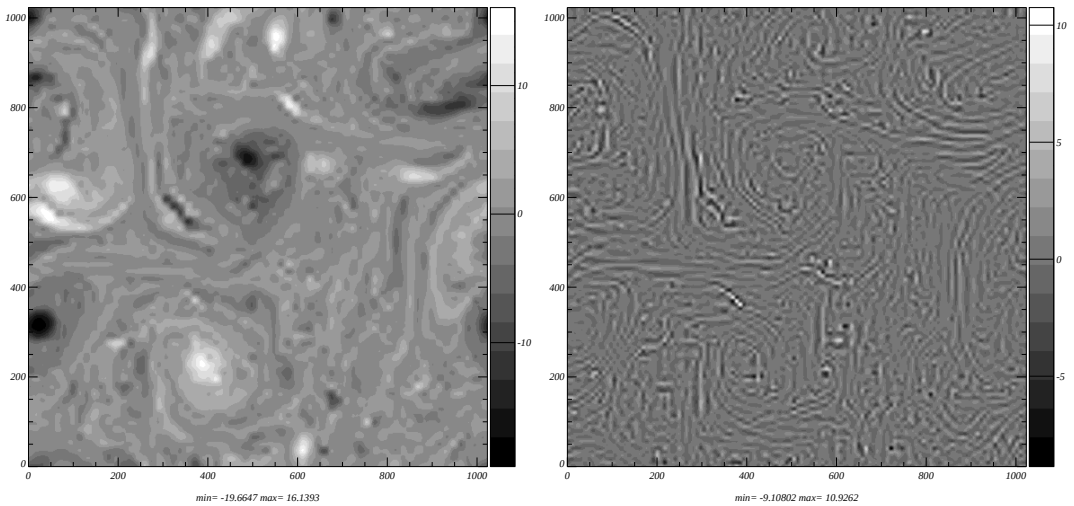


FIG. 1.4 – *champ projeté dans V_6 et W_6*

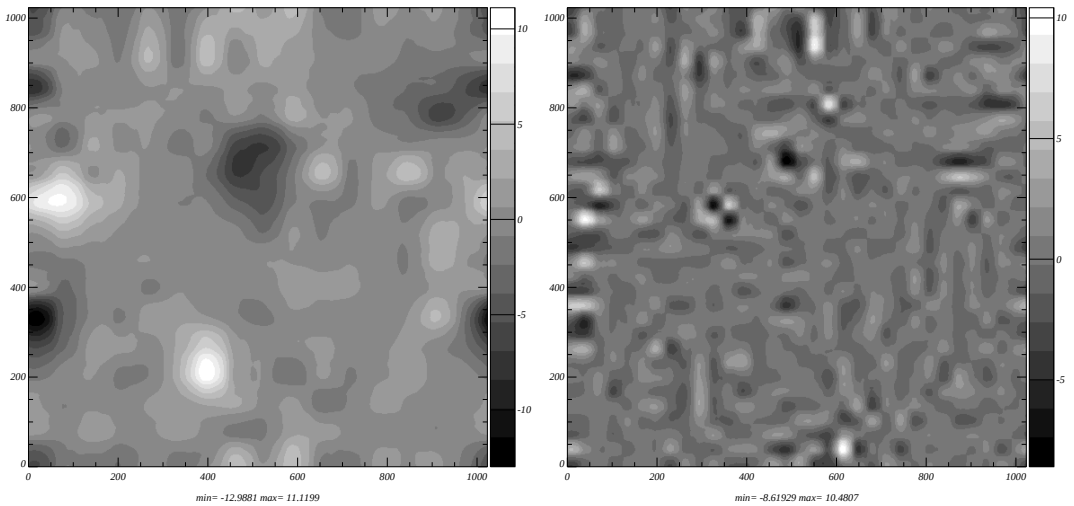


FIG. 1.5 – *champ projeté dans V_4 et W_4*

1.1.5 Interprétation de la Transformée en Ondelettes Discrète (TOD)

L'analyse en ondelettes périodiques peut être vue comme agissant sur un sous-espace de $L^2(\mathbb{T})$ ou comme une isométrie de $\mathbb{R}^n$.

Dans la section 1.1.3, partant d'une fonction f de $L^2(\mathbb{T})$, on définit $\tilde{f}$ sa projection orthogonale sur un sous-espace V_p. $\tilde{f}$ est repérée par ses coordonnées s_k^p, pour $0 \leq k < 2^p$ suivant la base des φ_k^p de V_p, s_k^p est donné par la formule $s_k^p = \int_0^1 f(x)\ \varphi_k^p(x)\ dx = \int_0^1 \tilde{f}(x)\ \varphi_k^p(x)\ dx$. Ensuite récursivement en décomposant V_{j+1} en la somme de W_j et V_j, on définit les coefficients d'ondelettes d_k^j, pour $0 \leq k < 2^j$ suivant la base des ψ_k^j de W_j et les coefficients d'échelle s_k^j, pour $0 \leq k < 2^j$ suivant la base des φ_k^j de V_j. Chacune de ces décompositions élémentaires correspond à une transformation orthogonale dans un espace de dimension finie 2^{j+1} dont la matrice est circulante avec sur les 2^j premières lignes le filtre g^{j+1} et sur les 2^j dernières le filtre h^{j+1}. La TOD fait donc passer des coefficients $(s_k^p)_k$ de $\tilde{f}$ sur la base $(\varphi_k^p)_k$ à ses coefficients $(d_k^j)_{j,k}$ sur la base $(\psi_k^j)_{j,k}$ et sa valeur moyenne s_0^0.

On peut aussi considérer la TOD comme une isométrie de $\mathbb{R}^{2^p}$, qui, partant d'un élément quelconque de $\mathbb{R}^{2^p}$ noté $x = (s_k^p;\ \ 0 \leq k < 2^p)$, lui associe l'élément $x' = Q_p x = (d_k^j;\ \ 0 \leq j \leq p-1,\ \ 0 \leq k < 2^j; s_0^0)$ de $\mathbb{R}^{2^p}$ par la succession de transformations orthogonales vues ci-dessus associées aux filtres g et h.

$$Q_p = Q_p^1 Q_p^2 Q_p^3 \cdots Q_p^p \tag{1.32}$$

$$Q_p^j = \left(\begin{array}{c|c} I_d^{2^p-2^j} & 0 \\ \hline 0 & A^j \end{array}\right)$$

avec A^j matrice d'ordre 2^j définie par : $A^j = \left(\begin{array}{c} g^j \\ \hline h^j \end{array}\right)$.

Cette isométrie correspond à un changement de base orthonormée dans $\mathbb{R}^{2^p}$. On pourra l'interpréter en termes de coefficients d'ondelettes d'une fonction en associant au vecteur $x = (s_k^p)$ la fonction $\tilde{f}$ de $V_p \subset L^2(\mathbb{T})$ définie par

$$\tilde{f}(x) = \sum_k s_k^p \varphi_k^p(x) = s_0^0 + \sum_{j,k} d_k^j \psi_k^j(x).$$

$$\tilde{f} = <\tilde{f}> + \sum_j \tilde{f}_j \quad \text{avec} \quad \tilde{f}_j \in W_j \quad \text{et} \quad \tilde{f}_j = \sum_k d_k^j \psi_k^j$$

On parlera dans la suite indifféremment de TOD d'une fonction ou de TOD d'un vecteur en référence à l'une et à l'autre de ces interprétations.

Dans notre étude, l'intérêt de la TOD repose sur l'hypothèse que le nombre de coefficients significatifs après transformation est moindre qu'avant transformation ou, autrement dit, que l'entropie d'information diminue par TOD. Cette propriété ne peut être appréciée qu'en utilisant l'interprétation en terme d'ondelettes, c'est à dire de fonctions de $L^2(\mathbb{T})$

régulières ou non. Donc quand on envisage la TOD d'une fonction, son interprétation est directe : elle nous est fournie par les théorèmes liant la décroissance des coefficients d'ondelettes à la régularité d'une fonction (section 1.2 et [Jaffard89, Holschneider et al.91, De Vore et al.92, Donoho93]) ; la décroissance est une puissance de l'échelle d'autant plus grande que la fonction est plus régulière localement. Si on envisage la TOD d'un vecteur de $\mathbb{R}^{2^p}$ on devra considérer la fonction de $V_p \in L^2(\mathbb{T})$ associée et déterminer si cette fonction est régulière ou non.

Il en va de même pour l'interprétation de la TOD en dimension deux ; on parlera aussi bien de TOD d'une fonction de deux variables que de TOD d'une matrice.

1.2 Représentation sur bases d'ondelettes de fonctions

Dans cette partie, nous supposerons toujours que la régularité des ondelettes est supérieure à celle de la fonction analysée.

1.2.1 Théorie

1.2.1.1 Décroissance des coefficients d'ondelettes en fonction de la régularité locale et globale

Commençons tout d'abord par donner un résultat dû à [Jaffard89, Jaffard et al.95] de décroissance des coefficients d'ondelettes pour les fonctions globalement Hölderiennes $C^s(\mathbb{R}^n)$, ; donc un résultat pour des fonctions globalement régulières, puis pour les fonctions localement Hölderiennes $C^s(x_0)$. Pour cela définissons cet espace pour s non entier :

Définition 3 *Soit $s > 0$, $s \notin \mathbb{N}$, f est élément de $C^s(x_0)$ si et seulement si il existe un polynôme $P_{x_0}[X]$ de degré inférieur à s tel que :*

$$|f(x) - P_{x_0}(x - x_0)| \leq C|x - x_0|^s \text{ dans un voisinage de } x_0 \tag{1.33}$$

f est $C^s(\mathbb{R}^n)$ si (1.33) est vérifiée pour tout x_0 de $\mathbb{R}^n$ et que C est indépendant de x_0.

Remarque : Si $s < 1$ alors $P_{x_0}(x) = f(x_0)$.

On a le théorème suivant :[1]

Théorème 1

$$f \in C^s(\mathbb{R}^n) \Leftrightarrow |d_{j,k}^{(i)}| \leq C 2^{-(\frac{n}{2}+s)j} \tag{1.34}$$

Avec $d_{j,k}^{(i)} = 2^{\frac{nj}{2}} \int f(x)\psi^{(i)}(2^j x - k)\, dx \quad (1 \leq i \leq n^2 - 1,\ j \in \mathbb{Z},\ k \in \mathbb{Z}^n)$.[2]

[1] Rappelons que dans cette partie, on suppose toujours que la régularité de l'ondelette est supérieure à celle de la fonction à analyser, donc ici que l'on utilise une AMR de régularité supérieure à la partie entière de $s + 1$.

[2] Les notations sont celles de la section 1.1.4.

Il existe également une version locale du théorème 1 ce qui permet d'exploiter pleinement les propriétés de localisation en espace des ondelettes :

Théorème 2

$$f \in C^s(x_0) \Rightarrow |d_{j,k}^{(i)}| \leq C 2^{-(\frac{n}{2}+s)j}(1+|2^j x_0 - k|)^s \tag{1.35}$$

Sur cette formule, on voit que la régularité de la fonction en un point se traduit par la décroissance en échelle des coefficients d'ondelettes proches de x_0 c'est à dire quand $|2^j x_0 - k|$ est petit.

On peut généraliser le théorème 1 à une classe d'espaces plus importante : les espaces de Besov (voir [Nikolskii05, Stein70]). Ceci va permettre d'unifier les résultats pour les espaces de Sobolev et pour les espaces de Hölder. En effet $H^s(\mathbb{R}^n) = B_2^{s,2}(\mathbb{R}^n)$ et $C^s(\mathbb{R}^n) = B_\infty^{s,\infty}(\mathbb{R}^n)$.

Les espaces de Besov ont été introduits pour la première fois comme espace d'interpolation entre les espaces de Sobolev ; ils sont définis en terme de L^p-module de continuité. Pour plus de simplicité, nous donnons les définitions pour la dimension un.

Pour $f \in L^p(\mathbb{R})$ et $h > 0$, nous introduisons le module de continuité :

$$\omega_p(f,h) = \|f(.+h) - f(.)\|_{L^p} \tag{1.36}$$

Ceci défini bien $\omega_p(f,h)$ car $\omega_p(f,h) \leq 2\|f\|_{L^p}$; remarquons aussi que pour $1 \leq p \leq \infty$, $\omega_p(f,h) \to 0$ quand $h \to 0$.

Une définition habituelle des espaces de Besov $B_p^{\alpha,q}$ pour $1 \leq p,q \leq \infty$ [3] est donnée par :

– Pour $0 < \alpha < 1$,

$$\begin{aligned} \text{si } q < \infty \quad B_p^{\alpha,q}(\mathbb{R}) &= \left\{ f \in L^p(\mathbb{R}) \ ; \ \int_0^{+\infty} [h^{-\alpha}\omega_p(f,h)]^q \frac{dh}{h} \ < \ \infty \right\}, \\ \text{si } q = \infty \quad B_p^{\alpha,\infty}(\mathbb{R}) &= \left\{ f \in L^p(\mathbb{R}) \ ; \ h^{-\alpha}\omega_p(f,h) \in L^\infty(\mathbb{R}_+^*) \right\}. \end{aligned}$$

$B_p^{\alpha,q}(\mathbb{R})$ est un espace de Banach associé à la norme :

$$\begin{aligned} \text{si } q < \infty \quad \|f\|_{B_p^{\alpha,q}} &= \|f\|_{L^p} + \left(\int_0^{+\infty} [h^{-\alpha}\omega_p(f,h)]^q \ \frac{dh}{h} \right)^{1/q}, \\ \text{si } q = \infty \quad \|f\|_{B_p^{\alpha,\infty}} &= \|f\|_{L^p} + \|h^{-\alpha}\omega_p(f,h)\|_\infty \ . \end{aligned}$$

– Pour $\alpha > 1$, α non entier,

[3] Cette définition peut être étendue pour $0 < p \leq 1$ (voir par exemple [De Vore et al.92])

$\alpha = [\alpha] + s, \quad 0 < s < 1$, ($[\alpha]$ est la partie entière de α),

$$B_p^{\alpha,q}(\mathbb{R}) = \left\{ f \in L^p(\mathbb{R}) \ ; \ f^{(m)} \in B_p^{s,q}(\mathbb{R}),\ 0 \leq m \leq [\alpha] \right\},$$

où $f^{(m)}$ est la dérivée m^{ieme} de f.

$B_p^{\alpha,q}(\mathbb{R})$ est un espace de Banach associé à la norme :

$$\|f\|_{B_p^{\alpha,q}} = \sum_{m=0}^{[\alpha]} \|f^{(m)}\|_{B_p^{s,q}} .$$

La caractérisation de ces espaces par les coefficients d'ondelettes est la suivante [Meyer90] :

Théorème 3 *f est dans $B_p^{s,q}(\mathbb{R}^n)$ si et seulement si*

$$\epsilon_j := 2^{j(\frac{n}{2}+s)} \left(\sum_{k\in\mathbb{Z}^n} \sum_{i=1}^{n^2-1} 2^{-nj} |d_{j,k}^{(i)}|^p \right)^{1/p} \quad \textit{est dans } l^q(\mathbb{Z}). \tag{1.37}$$

Et la norme $l^q(\mathbb{Z})$ des ϵ_j est une norme de l'espace $B_p^{s,q}(\mathbb{R}^n)$.

Si l'on prend $p = q = \infty$, on retrouve bien le théorème 1.
Si l'on prend $p = q = 2$, on a la caractérisation des espaces de Sobolev suivante :

Théorème 4 *f est dans $H^s(\mathbb{R}^n)$ si et seulement si*

$$\left(\sum_{j\in\mathbb{Z}} \sum_{k\in\mathbb{Z}^n} \sum_{i=1}^{n^2-1} 2^{2js} |d_{j,k}^{(i)}|^2 \right)^{1/2} < +\infty \tag{1.38}$$

De plus, on a l'équivalence de norme suivante :

$$\|f\|_{H^s(\mathbb{R}^n)} \sim \left(\sum_{j\in\mathbb{Z}} \sum_{k\in\mathbb{Z}^n} \sum_{i=1}^{n^2-1} 2^{2js} |d_{j,k}^{(i)}|^2 \right)^{1/2} < +\infty \tag{1.39}$$

Ce qui donne également le résultat pour $L^2(\mathbb{R}^n)$ (pour $s = 0$). Ces résultats montrent que l'erreur de projection dans un espace V_j décroît quand la régularité des fonctions augmente. Pour représenter des fonctions très régulières, il faudra donc utiliser des AMR d'une régularité supérieure à celle des fonctions analysées pour tirer pleinement parti de ces propriétés.

1.2.1.2 Approximation des fonctions par leur série d'ondelettes

Approximation linéaire A partir des résultats de la section précédente, nous allons donner une majoration de l'erreur lorsque l'on projette une fonction dans un espace V_j. Du théorème 4 on déduit facilement le

Corollaire 1

$$\|f - P_j f\|_{L^2(\mathbb{R}^n)} \leq C 2^{-js} \|f\|_{H^s(\mathbb{R}^n)} \tag{1.40}$$

où P_j est la projection orthogonale sur l'espace V_j.

Donc plus la fonction sera régulière, plus l'erreur de projection dans l'espace V_j sera faible en norme $L^2(\mathbb{R}^n)$. Ce résultat permet de justifier l'utilisation des bases d'ondelettes pour résoudre des équations aux dérivées partielles par des méthodes de Galerkin.

Approximation non linéaire Les espaces de Besov introduits précédemment vont nous permettre de donner un résultat d'approximation non linéaire tiré de [De Vore et al.92], c'est à dire une majoration de l'erreur quand on garde seulement un nombre fini de coefficients d'ondelettes d'une fonction, **les choix des coefficients étant adapté à la fonction**. Nous souhaitons remplacer la somme infinie :

$$f = \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}^n} \sum_{i=1}^{n^2-1} d_{j,k} \psi_{j,k}^{(i)} \tag{1.41}$$

par une somme finie. Notons Σ_N l'ensemble des fonctions comportant au plus N coefficients d'ondelettes et posons :

$$\sigma_N(f)_X := \inf_{S \in \Sigma_N} \|f - S\|_X \tag{1.42}$$

Remarque :

- $\sigma_N(f)_X$ **dépend de la base** dans laquelle on représente f.
- Si $X = L^2(\mathbb{R}^n)$, les $\psi_{j,k}^{(i)}$ formant une base orthonormée de $L^2(\mathbb{R}^n)$, ce minimum est atteint avec la fonction qui est la somme des N plus grandes composantes de f.

$\sigma_N(f)_X$ représente l'erreur d'approximation dans l'espace Σ_N en utilisant la norme de l'espace X. Si l'on prend comme espace d'approximation l'espace $L^p(\mathbb{R}^n)$, on a l'inégalité suivante :

$$\sigma_N(f)_{L^p(\mathbb{R}^n)} \leq C N^{-s/n} \|f\|_{B_\tau^{s,\tau}} \quad \text{avec} \quad \tau = (\frac{s}{n} + \frac{1}{p})^{-1} \tag{1.43}$$

Mais la relation entre τ et p ne permet pas une caractérisation de l'erreur par la norme dans les espaces classiques (H^s et C^s). Nous allons donc donner un autre résultat d'approximation non linéaire présenté dans [Donoho93]. Ce résultat donne une estimation de la décroissance des coefficients d'ondelettes. Le résultat est le suivant :

Théorème 5 *Soit X un sous-espace de $L^2(\mathbb{R}^n)$, posons $\sigma_N^*(X) = \sup_{f \in X} \sigma_N(f)_{L^2(\mathbb{R}^n)}$*

Si l'on prend pour ensemble X les fonctions à variations bornées, on a :

- *Dans la base de Haar :* $\sigma_N^\star(BV) \asymp \frac{1}{N} \quad N \to \infty$
- *Alors que dans la base de Fourier :* $\sigma_N^\star(BV) \asymp \frac{1}{\sqrt{N}} \quad N \to \infty$

On voit donc que si l'on utilise uniquement N coefficients pour représenter une fonction à variation bornée, on fera une erreur bien inférieure dans la base de Haar par rapport à la base de Fourier.

Ce résultat peut s'étendre à des fonctions plus régulières en utilisant des ondelettes de régularité adaptée à celle des fonctions (*voir* [Donoho93]). D'autres résultats de compressions sont donnés dans [Meyer90, Dahmen et al.94].

Les résultats de cette partie justifient l'emploi de méthode de type Galerkin en sélectionnant uniquement certains vecteurs de la base adaptés à la solution pour résoudre des équations aux dérivées partielles.

1.2.2 Exemples

Dans cette partie, nous allons analyser plusieurs champs à l'aide de différentes sortes d'ondelettes, et comparer les taux de compressions obtenus avec ceux d'une analyse de Fourier.

Commençons tout d'abord par donner une représentation dans l'espace physique des champs que nous allons analyser en ondelettes et par transformation de Fourier (figures 1.6 à 1.8). Ces champs sont issus de la simulation numérique des équations de Navier-Stokes bidimensionnelles incompressibles sur un domaine carré bipériodique utilisant une méthode pseudo-spectrale [Babiano et al.87, Basdevant et al.81]. Sur ces représentations on constate que de tels champs sont composés de structures de différentes échelles, généralement assez bien isolées les unes des autres. Ceci va permettre de représenter ces champs efficacement sur des bases d'ondelettes.

Nous présentons ensuite (figure 1.9-1.10) le pourcentage d'énergie conservée des champs en fonction du pourcentage de degrés de liberté utilisés en utilisant une analyse en ondelette et une analyse de Fourier. Sur les figures de gauche, nous comparons les taux de compressions obtenus à l'aide d'une analyse en ondelettes avec ceux obtenus par une analyse de Fourier. Sur les figures de droite, nous représentons le nombre de degrés de liberté utilisés pour représenter le champ dans chaque échelle et le nombre total de degrés de liberté utilisés. L'échelle la plus petite est numérotée 1, elle représente au total 75 % des degrés de liberté possibles.

1.2.2.1 Description des champs

initialfred et dalila La résolution est de 128^2, et l'écoulement est forcé en Fourrier à $k = 10$ en gardant ce mode constant au cours du temps. Une hyperviscosité est utilisée avec un Laplacien itéré 8 fois[4] afin de limiter l'effet de la dissipation sur les échelles

[4] dans l'équation de Navier-Stokes, on remplace Δ par Δ^8.

intermédiaires. Un désaliasage[5] partiel est obtenu en tronquant à $k_t = 0.9\, k_{max}$ (avec $k_{max} = 64$). L'état initial est un champ aléatoire (module et phase aléatoires). Ce champ provient d'une longue intégration, au bout de laquelle l'écoulement est bien stabilisé, au sens où les grandeurs spectrales (spectres d'énergie, flux) ne bougent plus significativement au cours du temps. **initialfred** correspond au premier pas de temps où le champ est considéré comme stabilisé, **dalila** à un temps supérieur (origine : A. Babiano [Babiano et al.87]).

f10a1n10-20-25-40 La résolution est de 1024^2, et l'écoulement est non forcé. Une hyperviscosité est utilisée avec un Laplacien itéré 2 fois. L'état initial est un champ aléatoire (module et phase aléatoires). Le champ est représenté aux temps 10, 20, 25, et 40, le pas de temps est de $\delta t = 3.125\ 10^{-4}$ (origine : B. Legras [Santangelo et al.89]).

1.2.2.2 Analyse des résultats

initialfred et dalila Sur la figure 1.9, on constate que la base de Fourier permet une représentation légèrement plus lacunaire que la base d'ondelettes. Ceci s'explique par deux faits :

1. Le taux de compression en ondelettes comme en Fourier est d'autant plus important que la résolution est grande. Mais ce taux augmente plus vite sur la base d'ondelettes, 128^2 est donc la résolution à laquelle les deux représentations comportent sensiblement le même nombre de coefficients ; pour des résolutions plus faibles la base de Fourier est meilleure, pour des résolutions plus fortes la représentation est plus lacunaire en ondelettes.
2. Dans le code utilisé pour ces simulations, on effectue une troncature en Fourier pour éviter les phénomènes d'aliasing, un certain nombre de coefficients de Fourier sont donc imposés nuls, ce qui réduit le nombre de modes de Fourier nécessaires pour représenter le champ.

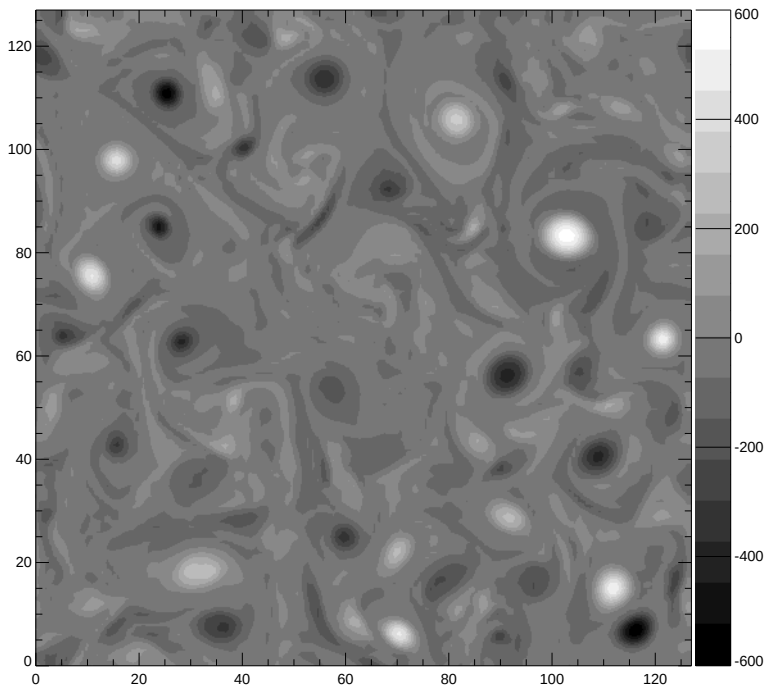
Sur ce champ forcé, on constate que le nombre de coefficients nécessaires pour représenter un certain pourcentage d'énergie varie peu en fonction du temps.

f10a1n10-20-25-40 Sur les figures 1.9-1.10, on constate que pour un champ à haute résolution (1024^2) il faut beaucoup moins ($\approx$ deux fois moins) de coefficients d'ondelettes que de coefficients de Fourier pour conserver une énergie donnée. Ceci est valable quel que soit le pourcentage d'énergie conservée. Sur cette simulation non forcée, on constate qu'en Fourier comme en ondelettes, le nombre de degrés de liberté nécessaire à la représentation du champ décroit au cours du temps. En effet, la condition initiale aléatoire ne possède aucune régularité, elle va donc nécessiter de nombreux coefficients d'ondelettes pour une représentation acceptable. Puis le champ s'organise en structures cohérentes qui sont bien représentées en ondelettes.

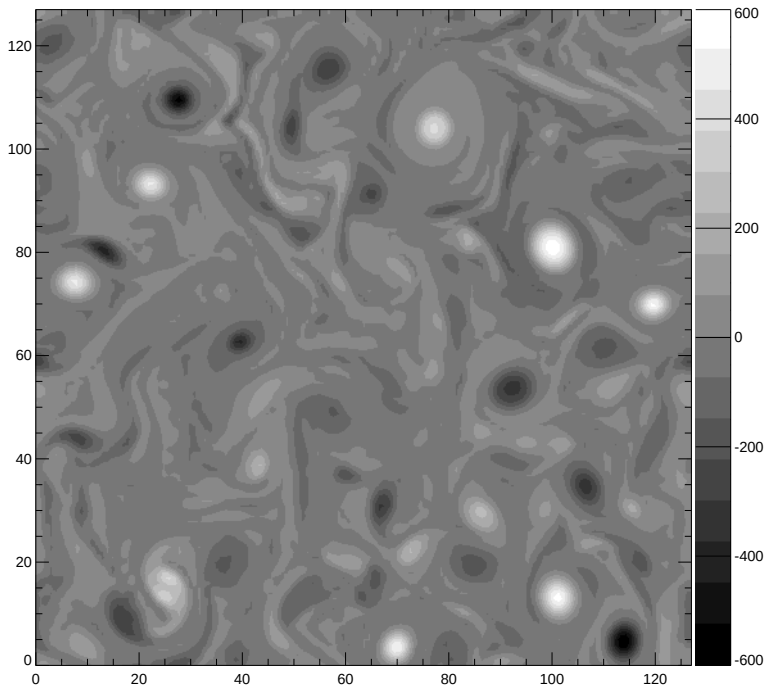
[5] Méthode qui consiste à mettre à zéro les modes de Fourrier élevé (ici, compris dans l'intervalle $[0.9\ k_{max}, k_{max}]$) afin d'éviter un recouvrement du spectre lors du calcul du terme quadratique dans l'espace physique.

Au temps $T = 40$, la solution composée de quelques tourbillons nécessite seulement 2% de coefficients pour conserver 98% de l'énergie.

Sur les courbes détaillant le nombre de coefficients d'ondelettes conservés échelle par échelle (figures de droite dans les figures 1.9-1.10), on constate que la compression est essentiellement effectuée sur les petites échelles. En effet, pour les grandes échelles, les courbes sont horizontales : quel que soit le pourcentage d'énergie conservée, on utilise tous les coefficients qui représentent une grande part de l'énergie du champ. Alors que pour les petites échelles, on a une très forte variation du nombre de degrés de liberté notamment pour représenter toute l'énergie du champ (la variation du nombre de degrés de liberté utilisés est très importante entre 99.5 % et 100 %). Ce fait s'explique facilement : les petites échelles représentent la majorité des coefficients pour seulement une petite fraction de l'énergie. Par exemple, l'échelle la plus petite représente 75 % des coefficients pour moins de 1 % d'énergie pour tous les champs 1024^2 (de l'ordre de 3 % pour les champs 128^2). De plus cette énergie est localisée sur seulement quelques coefficients d'ondelettes (voir figure 1.3 page 19) à cause de l'organisation spatiale des champs turbulents.

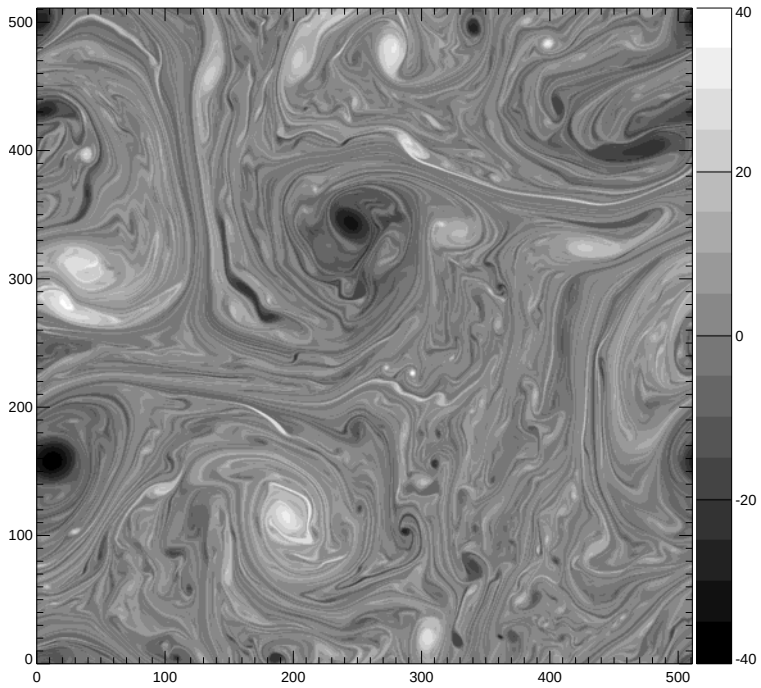


(a) dalila

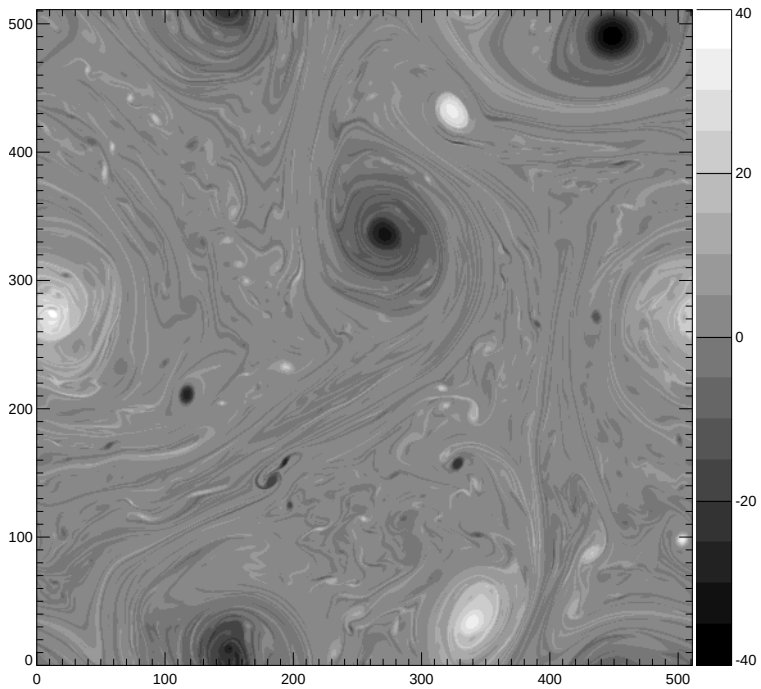


(b) initialfred

FIG. 1.6 – *Champs de vorticité*

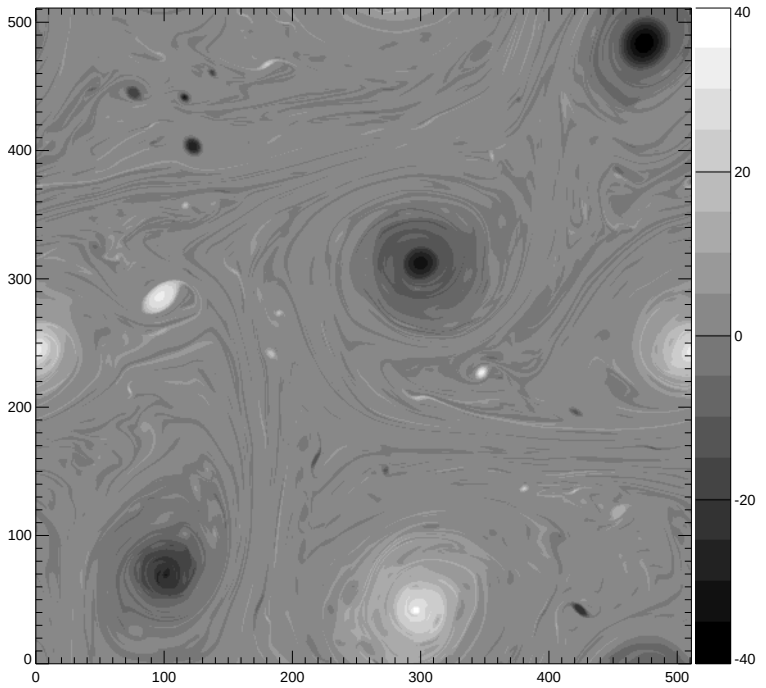


(a) f10a1n10

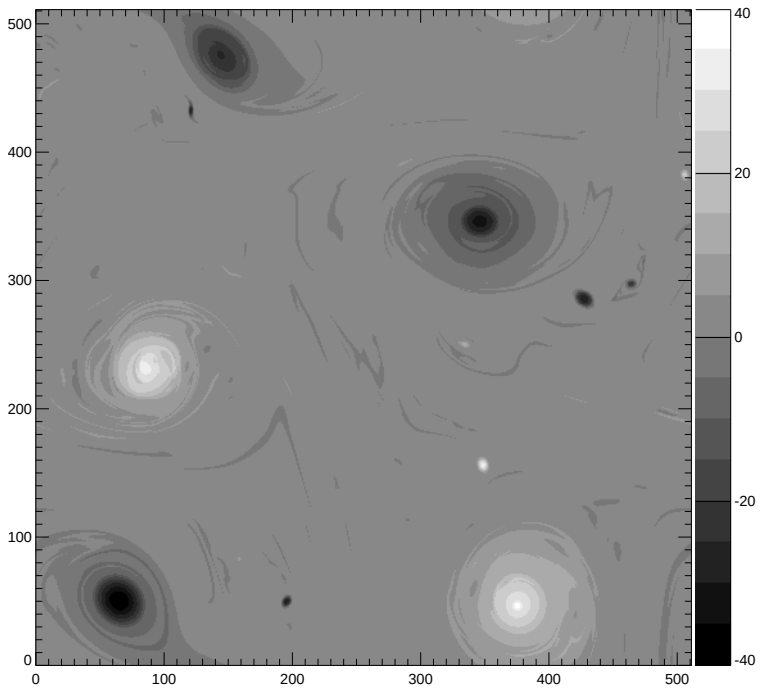


(b) f10a1n20

FIG. 1.7 – *Champs de vorticité*



(a) f10a1n25



(b) f10a1n40

FIG. 1.8 – *Champs de vorticité*

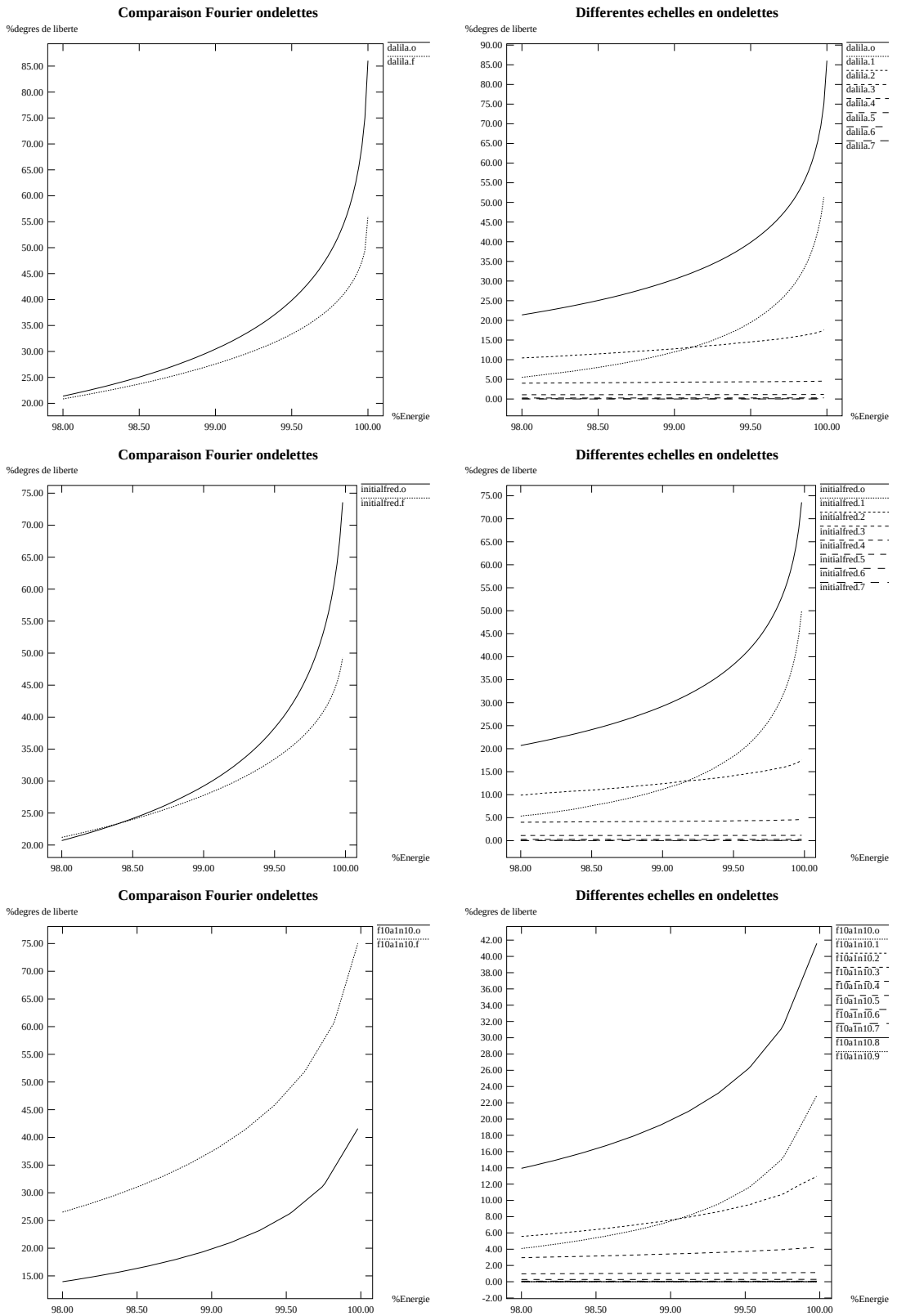


FIG. 1.9 – *Taux de compression par analyse en Fourier et en ondelettes échelle par échelle*

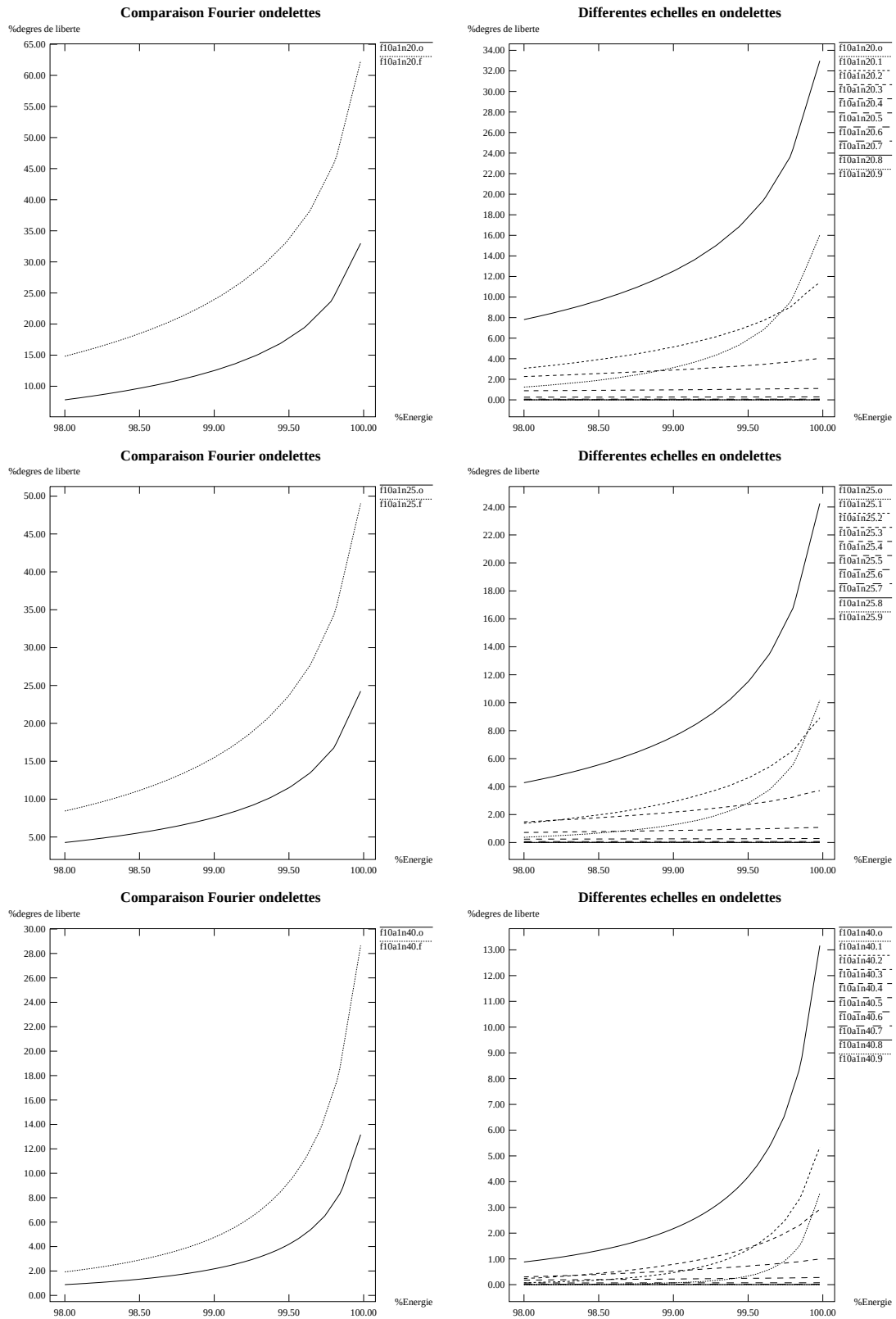


FIG. 1.10 – *Taux de compression par analyse en Fourier et en ondelettes échelle par échelle*

1.3 Représentation sur bases d'ondelettes d'opérateurs

1.3.1 Théorie

Comme dans la section 1.2.1, nous allons commencer par donner des résultats théoriques de compression par ondelettes, mais cette fois pour des opérateurs. Pour cela introduisons les opérateurs de Calderón-Zygmund :

Définition 4 *Les opérateurs intégraux continus sur $L^2(\mathbb{R})$, dont le noyau K vérifie les conditions :*

$$|K(x,y)| \leq \frac{1}{|x-y|^M}, \tag{1.44}$$

$$|\partial_x^M K(x,y)| + |\partial_y^M K(x,y)| \leq \frac{C}{|x-y|^{1+M}}, \tag{1.45}$$

sont appelés opérateurs de Calderón-Zygmund d'ordre M (voir [Meyer90]).

Nous cherchons à appliquer un opérateur T à une fonction u, soit à calculer $T(u) = f$, où T est un opérateur intégral unidimensionnel, qui s'écrit sous la forme : $T(u)(x) = \int_{\mathbb{R}} K(x,y)u(y)\,dy$.

Pour discrétiser un tel opérateur, il suffit de discrétiser son noyau. On se ramène ainsi au cas des champs bidimensionnels sur lesquels on peut faire deux types d'analyses : une analyse tensorielle ou une AMR.

Une analyse tensorielle, c'est à dire que l'on décompose le noyau sur une base formée des produits tensoriels d'ondelettes unidimensionnelles ; une telle base est de la forme : $(\psi_k^j(x)\psi_{k'}^{j'}(y))$. L'opérateur T est donc représenté par la matrice :

$$d_{k,k'}^{j,j'} = \iint_{\mathbb{R}^2} K(x,y)\psi_k^j(x)\psi_{k'}^{j'}(y)\;dxdy, \tag{1.46}$$

ce qui n'est rien d'autre que :

$$d_{k,k'}^{j,j'} = \langle\, T(\psi_k^j) \mid \psi_{k'}^{j'} \,\rangle \tag{1.47}$$

Cette décomposition est appelée la décomposition standard d'un opérateur.

Pour ce type de décomposition, nous avons le théorème suivant, tiré de [Meyer90] :

Théorème 6 *Si la base d'ondelettes a M moments nuls et si K est le noyau d'un opérateur de Calderón-Zygmund d'ordre M, alors ses coefficients d'ondelettes vérifient :*

$$|d_{k,k'}^{j,j'}| \leq C\; 2^{-|j-j'|(M+\frac{1}{2})} \left(\frac{2^{-j}+2^{-j'}}{2^{-j}+2^{-j'}+|k2^{-j}-k'2^{-j'}|} \right)^{1+M} \tag{1.48}$$

On a deux types de décroissances : d'une part, comme au théorème 7, on a une décroissance des coefficients à partir de la diagonale de chaque bloc constituant la matrice, d'autre part, les blocs eux-mêmes sont formés de coefficients décroissants lorsque l'on s'éloigne de la diagonale de la matrice globale, c'est à dire, lorsque les deux échelles d'analyse en x et en y du bloc deviennent très différentes.

On peut également utiliser une AMR bidimensionnelle présentée à la section 1.1.4 ; la base correspondante est alors composée de trois ondelettes génératrices différentes, les coefficients associés étant appelés $\alpha_{i,l}^j$, $\beta_{i,l}^j$ et $\gamma_{i,l}^j$ (un indice d'échelle j et deux indices de position i (suivant les x) et l (suivant les y). Donnons par exemple l'expression de $\alpha_{i,l}^j$:

$$\alpha_{i,l}^j = \iint_{\mathbb{R}^2} K(x,y)\psi_i^j(x)\psi_l^j(y)\ dxdy \tag{1.49}$$

Le résultat de compression est cette fois dû à [Beylkin et al.91] :

Théorème 7 *Si la base d'ondelettes a M moments nuls, et si K est un noyau d'opérateur de Calderón-Zygmund d'ordre M, alors les matrices α^j, β^j et γ^j (1.28)-(1.30) satisfont l'inégalité :*

$$|\alpha_{i,l}^j| + |\beta_{i,l}^j| + |\gamma_{i,l}^j| \leq \frac{C_M}{1+|i-l|^{M+1}}, \tag{1.50}$$

pour tout $|i-l| \geq 2M$.

Ce théorème signifie que la décroissance à partir de la diagonale des coefficients dans les blocs représentant le noyau en base d'ondelettes (figure 1.1) est d'autant plus importante que le noyau est régulier.

Le premier type d'analyse (analyse tensorielle) est la décomposition standard appelée aussi analyse en ondelettes dans la suite du mémoire, le deuxième type d'analyse est la décomposition non standard, point de départ de la méthode BCR présentée à la section 2.1.3. Dans la pratique, ces deux représentations conduisent à des représentations « lacunaires » du noyau $K(x,y)$ associé à un opérateur Calderón-Zygmund.

1.3.2 Exemples

Un exemple d'opérateur de Calderón-Zygmund est l'opérateur de la chaleur, c'est à dire l'opérateur $T(u) = (1-\nu\frac{\partial^2}{\partial x^2})^{-1}u$ associé à l'équation aux dérivées partielles suivante :

$$\frac{\partial}{\partial t}u - \nu\frac{\partial^2}{\partial x^2}u = f, \qquad x \in \mathbb{T} \tag{1.51}$$

Cette équation est d'abord discrétisée en temps par un schéma d'Euler implicite :

$$(1-\nu\delta t\frac{\partial^2}{\partial x^2})u^{n+1} = u^n + \delta t f \tag{1.52}$$

puis en espace par un schéma aux différences finies d'ordre 4. On a donc à résoudre un système de la forme :

$$AU^{n+1} = U^n + F \tag{1.53}$$

Nous allons à présent donner trois représentations différentes de la matrice A (associée à l'opérateur T^{-1}) et de son inverse A^{-1} (associée à l'opérateur T) : une représentation dans l'espace physique (fig. 1.11), une analyse en ondelettes tensorielles (fig. 1.12) et une analyse en ondelettes bidimensionnelles (fig. 1.13) . A^{-1} est un exemple de discrétisation d'opérateur de Calderón-Zygmund ; nous utiliserons essentiellement cet opérateur dans la partie numérique, nous montrerons également qu'à l'aide de méthodes de directions alternées, on peut résoudre des problèmes multidimensionnels à partir d'opérateurs unidimensionnels.

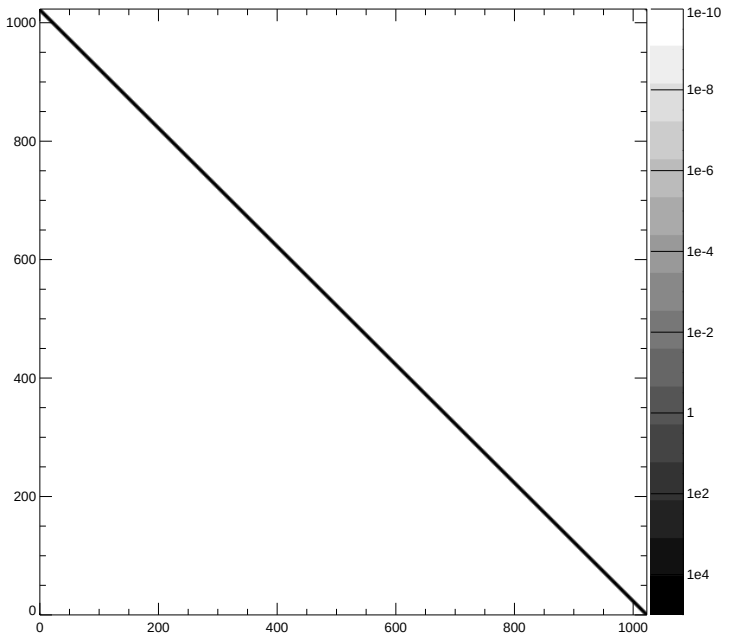
Sur ces figures, on constate que, dans l'espace physique, l'opérateur direct fig. 11(a) est beaucoup plus creux que son inverse fig. 11(b). Lors de l'utilisation d'un schéma implicite, si l'on reste dans l'espace physique, le système linéaire sera donc résolu par une méthode itérative faisant intervenir l'opérateur direct. En effet, la matrice de l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})$ en différences finies est diagonale-bande (plus les effets de la périodisation), il est donc très économique de l'appliquer à un vecteur : le nombre d'opérations à effectuer est de cN opérations où N est la taille du vecteur et c la largeur de bande de la matrice.

Par contre, sur une base d'ondelettes tensorielles ou bidimensionnelles, la structure et le nombre de coefficients significatifs dans la matrice de l'opérateur direct fig. 12(a)-13(a) et dans la matrice de son inverse fig. 12(b)-13(b) sont équivalents (voir figure 1.14). On a un nombre similaire de coefficients significatifs dans les matrices pour ces deux types de décomposition.

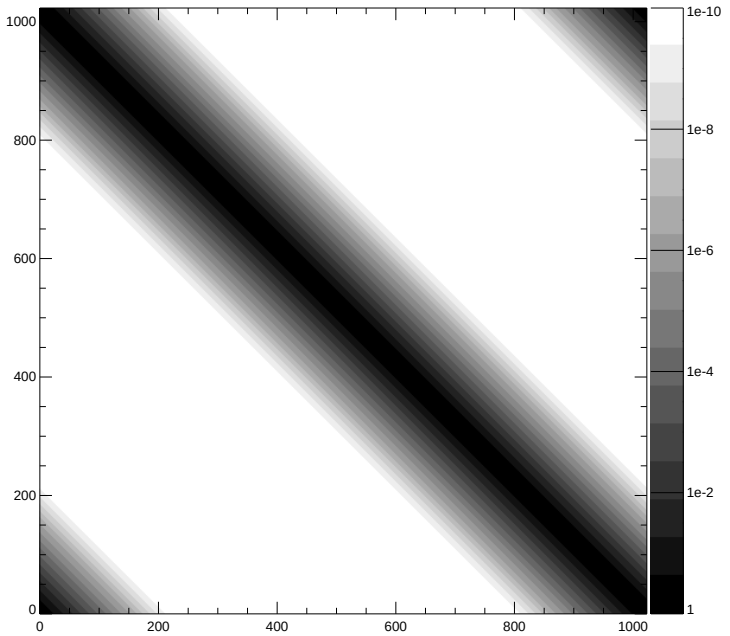
La matrice est initialement discrétisée par un schéma aux différences finies d'ordre quatre pour la décomposition en ondelettes tensorielles, et d'ordre deux pour la décomposition en ondelettes bidimensionnelles : seule la largeur de bande pour la matrice de l'opérateur de dérivation varie (la largeur de bande de cet opérateur est proportionnelle à l'ordre du schéma utilisé dans le cas de l'utilisation d'ondelettes à support compact). Le nombre de coefficients significatifs dans la matrice de l'opérateur inverse varie peu. Ainsi, si l'on effectue une factorisation LU de la matrice A décomposée comme sur la figure 12(a), on obtient une matrice complètement pleine : dans ce cas, pour résoudre le système linéaire (1.53), il sera plus intéressant d'inverser la matrice une fois pour toute et d'appliquer son inverse au second membre de l'EDP plutôt que d'utiliser une méthode itérative (matrice ayant un nombre comparable de coefficients, mais qu'il faut appliquer plusieurs fois) ou une méthode directe, (par exemple une factorisation LU, factorisation plus pleine que l'inverse).

En résumé, pour résoudre le système (1.53), le plus efficace est d'utiliser une méthode itérative dans l'espace physique, et de calculer l'inverse de A et de l'appliquer à $U^n + F$ en ondelettes.

Sur la figure 1.14, on représente le nombre de coefficients d'ondelettes supérieurs à 10^{-8} dans l'opérateur direct (correspondant au schéma d'Euler explicite) et son inverse (correspondant au schéma d'Euler implicite) en fonction de l'ordre de l'ondelette. On

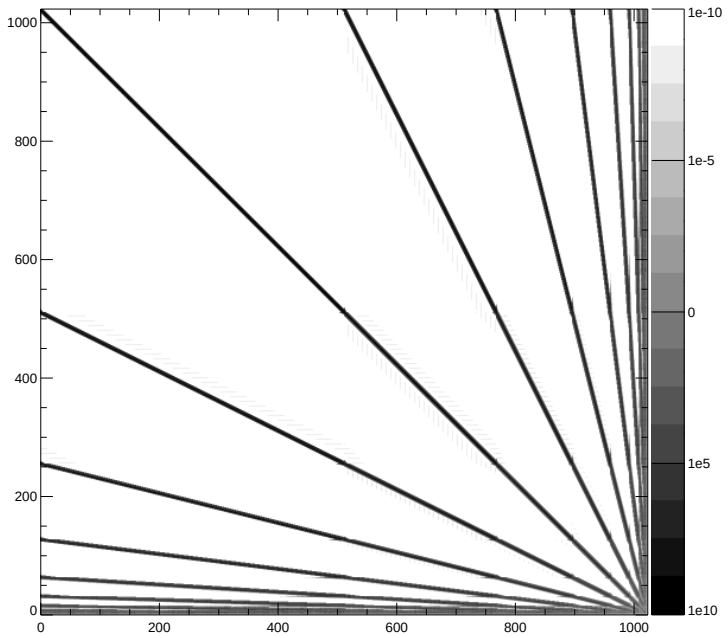


(a) Matrice de l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})$ en différences finies d'ordre 4, (taille : 1024^2, $\alpha = 10^{-4}$)

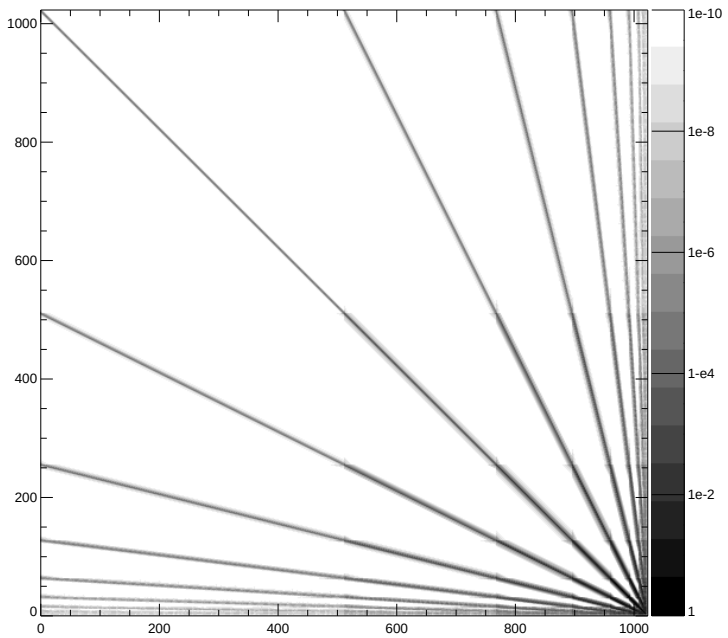


(b) Matrice l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})^{-1}$ en différences finies d'ordre 4,(taille : 1024^2, $\alpha = 10^{-4}$)

FIG. 1.11 – *Opérateurs en différences finies*

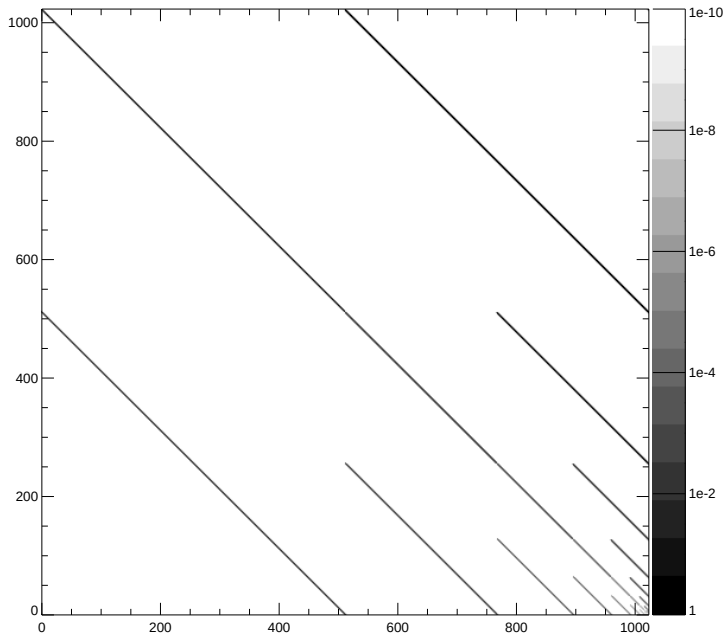


(a) Forme standard de la matrice de l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})$ en différences finies d'ordre 4, (taille : 1024^2,$\alpha = 10^{-4}$)

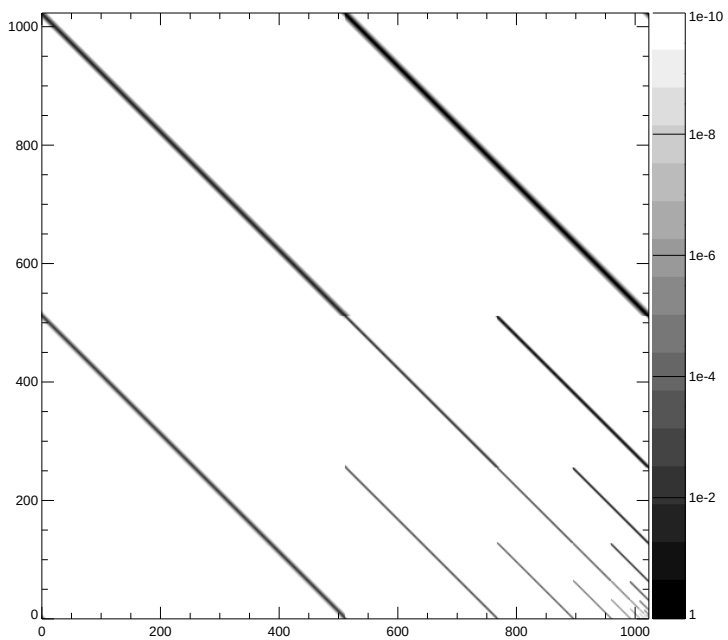


(b) Forme standard de la matrice de l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})^{-1}$ en différences finies d'ordre 4, (taille : 1024^2,$\alpha = 10^{-4}$)

FIG. 1.12 – *Opérateurs analysés en ondelettes tensorielles (méthode standard)*



(a) Forme non standard de la matrice de l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})$ en différences finies d'ordre 2, (taille : 1024^2,$\alpha = 10^{-4}$)



(b) Forme non standard de la matrice de l'opérateur $(1 - \alpha \frac{\partial^2}{\partial x^2})^{-1}$ en différences finies d'ordre 2, (taille : 1024^2,$\alpha = 10^{-4}$)

FIG. 1.13 – *Opérateurs analysés en ondelettes 2D (méthode non standard)*

constate que pour l'opérateur direct, la matrice se remplit lorsque l'on augmente l'ordre de l'ondelette. Ceci s'explique simplement : cet opérateur est très localisé dans l'espace physique (figure 11(a)), et le support des ondelettes est proportionnel à leur ordre (deux fois l'ordre dans le cas des ondelettes de Daubechies utilisées ici), donc le nombre de coefficients d'ondelettes est proportionnel à l'ordre.

Par contre, son inverse étant régulier, sa représentation est d'autant plus creuse que l'ordre de l'ondelette analysatrice est important (théorème 6) ; ceci explique la diminution du nombre de coefficients par lignes constaté entre l'ordre un et l'ordre cinq. Comme il est également localisé en espace (figure 11(b)), à partir de l'ordre cinq ces deux phénomènes se compensent.

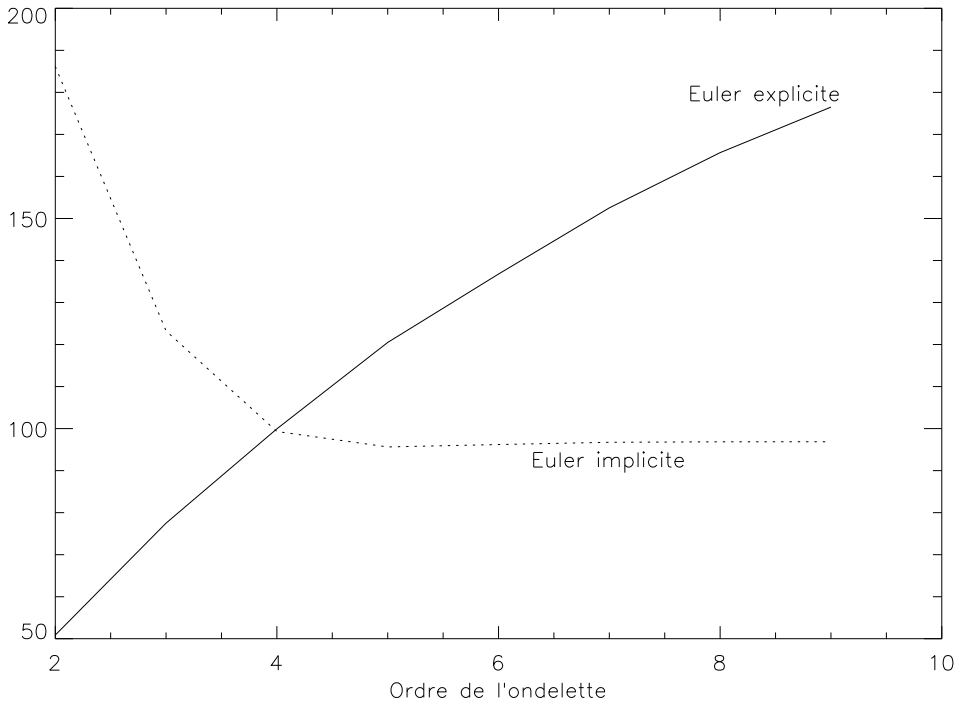


FIG. 1.14 – *Nombre moyen par ligne d'éléments supérieurs au seuil* $\epsilon = 10^{-8}$ *dans la matrice utilisée dans les schémas d'Euler explicite et implicite pour l'équation de la chaleur* ($\nu \delta t = 10^{-4}$) *en base d'ondelettes en fonction de l'ordre de celles-ci*

Chapitre 2

Produits rapides de matrices

Dans cette partie nous allons montrer comment exploiter la compression des matrices et des champs mono et bidimensionnels apportée par la transformée en ondelettes discrète, pour effectuer rapidement des produits matrice-vecteur et des produits de matrices. Pour calculer le produit $Y = BX$, nous allons effectuer un changement de base pour nous placer dans une base d'ondelettes et calculer le produit BX de manière approchée : les coefficients de la matrice B inférieurs à un seuil ϵ_B sont mis à 0, de même que ceux du vecteur X inférieurs à un autre seuil $\epsilon_X \|X\|$.

2.1 Transformations du système linéaire $Y = BX$

On se place dans l'espace $\mathbb{R}^N$ et on désire calculer le produit $Y = BX$. Le but de cette section est d'étudier différentes décompositions permettant de rendre la matrice B et les vecteurs X , Y creux, de façon à optimiser le coût du produit matrice-vecteur BX. Les méthodes présentées dans cette section ont déjà fait l'objet d'études théoriques dans [Beylkin et al.91, Dahmen et al.94].

2.1.1 Analyse sur la base de Fourier

Cette analyse (classique) s'applique seulement quand la matrice B est circulante (c'est à dire issue d'un opérateur invariant par translation) ; dans ce cas le produit $Y = BX$ s'écrit simplement comme un produit de convolution :

$$y_n = \sum_{i=0}^{N-1} b_{n-i} \cdot x_i \tag{2.1}$$

Si l'on a posé :

$$B = (b_{i-j})_{i,j=0,N-1} \ , \ Y = (y_i)_{i=0,N-1} \text{ et } X = (x_i)_{i=0,N-1} \tag{2.2}$$

les b_i étant périodiques de période N.

Dans la base de Fourier, ce produit de convolution devient un simple produit terme à terme :

$$\widehat{y}_k = N\hat{b}_k \cdot \hat{x}_k,$$

$\hat{X} = (\hat{x}_k)_{k=0,N-1}$ étant la transformée de Fourier discrète de X : $\hat{x}_k = \frac{1}{N}\sum_{j=0}^{N-1} x_j e^{-i\frac{2\pi jk}{N}}$, et $(\hat{b}_k)$ la transformée de Fourier discrète des (b_j). Dans cette base, la matrice B est transformée en une matrice $\hat{B}$ diagonale :

$$\hat{B} = \Big(diag(\hat{b}_k)\Big)_{k=0,N-1} \tag{2.3}$$

Par contre, en général, $\hat{X}$ n'est pas un vecteur creux ; pour qu'il soit creux il faut que X soit la superposition d'un petit nombre d'ondes pures.

2.1.2 Analyse en ondelettes – Forme standard

Cette méthode présentée dans [Beylkin92] consiste à effectuer une TOD du système linéaire. On se ramène à résoudre $Y' = B'X'$ avec (Q_p est la matrice de la TOD, équation (1.32) introduite à la section 1.1.5) :

- $Y' = Q_p Y$ la TOD de Y
- $X' = Q_p X$ la TOD de X
- $B' = Q_p B Q_p^{-1}$

Les vecteurs Y' et X' seront creux si les fonctions associées sont régulières au sens des ondelettes (voir section 1.1.5).

la matrice B' quant à elle a une structure bloc illustrée sur la figure 2. Elle est composée de p^2 blocs $B^{i,j}(0 \leq i,j \leq p-1)$ le bloc $B^{i,j}$ opérant de W_j dans W_i.

Si on interprète le système linéaire en terme de fonctions de V_p (voir section 1.1.5), X étant, dans la base des fonctions d'échelle de V_p, associé à la fonction f, Y à la fonction u et B à l'opérateur $\mathcal{B}$, alors B' est la matrice de l'opérateur $\mathcal{B}$ dans la base d'ondelettes de V_p, ainsi :

$$B'^{i,j}_{k,l} = \int_T \psi_0^i(x - 2^{-i}k)\mathcal{B}\psi_0^j(x - 2^{-j}l)dx \ , \ 0 \leq k \leq 2^i - 1 \ , \ 0 \leq l \leq 2^j - 1 \tag{2.4}$$

Pour des matrices issues de la discrétisation d'un noyau de Green, la double localisation des bases d'ondelettes (espace et échelle) implique une structure bande pour les blocs $B^{i,j}$ (décroissance en espace), et une décroissance des blocs quand on s'écarte de la diagonale de B' (décroissance en échelle) (Théorème 6 section 1.2.1 ; ce qui est confirmé par les figures 2.2 et 2.3). Pour les exemples numériques tirés de la mécanique des fluides, les vecteurs X et Y seront très bien représentés en ondelettes, dans le sens où peu de coefficients d'ondelettes seront non négligeables (voir section 1.2). Ainsi le produit $Y = BX$ nécessitera peu d'opérations.

$B^{3,3}$	$B^{3,2}$	$B^{3,1}$	$B^{3,0}$
$B^{2,3}$	$B^{2,2}$	$B^{2,1}$	$B^{2,0}$
$B^{1,3}$	$B^{1,2}$	$B^{1,1}$	$B^{1,0}$
$B^{0,3}$	$B^{0,2}$	$B^{0,1}$	$B^{0,0}$

FIG. 2.1 – *Structure de la matrice B' en base d'ondelettes, chaque bloc $B^{i,j}$ correspond à $B^{i,j} = \Pi_{W_i} B \Pi_{W_j}$*

2.1.3 Analyse de Beylkin, Coifman et Rokhlin (BCR) – Forme non standard

Une troisième méthode est présentée par Beylkin, Coifman et Rokhlin dans [Beylkin et al.91]. Elle consiste non pas à effectuer un changement de base, mais à effectuer une TOD bidimensionnelle de la matrice B. Si B est associée à un noyau de Green, on s'attend à ce que sa TOD conduise à une matrice ayant très peu de coefficients significatifs, c'est à dire une matrice très creuse (théorème 7 de la section 1.3.1 confirmé par les figures 2.5, 2.6).

Un réarrangement astucieux de la matrice TOD de B ainsi qu'une double représentation du vecteur X suivant les sous-espaces W_j et V_j, c'est à dire les coefficients d'ondelettes et les coefficients d'échelles, permettent le calcul (rapide !) du produit $Y = BX$.

L'interprétation de l'analyse non-standard est plus aisée en termes de fonctions et d'opérateurs qu'en termes de vecteurs et matrices. Soient u_p, f_p et $\mathcal{B}_p$ les fonctions et l'opérateur associés (au sens de la section 1.1.5) dans V_p aux vecteurs Y et X et à la

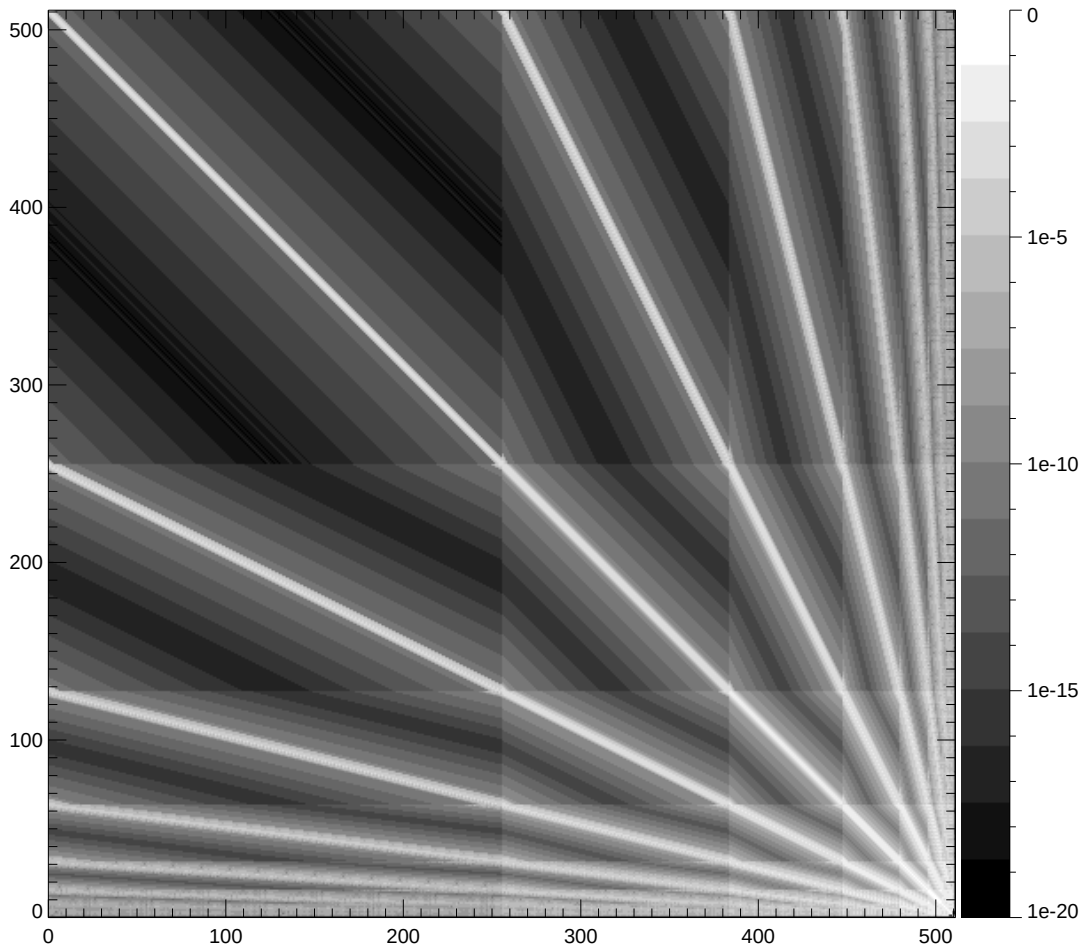


FIG. 2.2 – *Représentation de la matrice B en ondelettes (à support compact d'ordre 4) où B est la matrice de l'opérateur : $\mathcal{B} = (1 - \nu\delta t\partial^2_{xx})^{-1}$ (échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

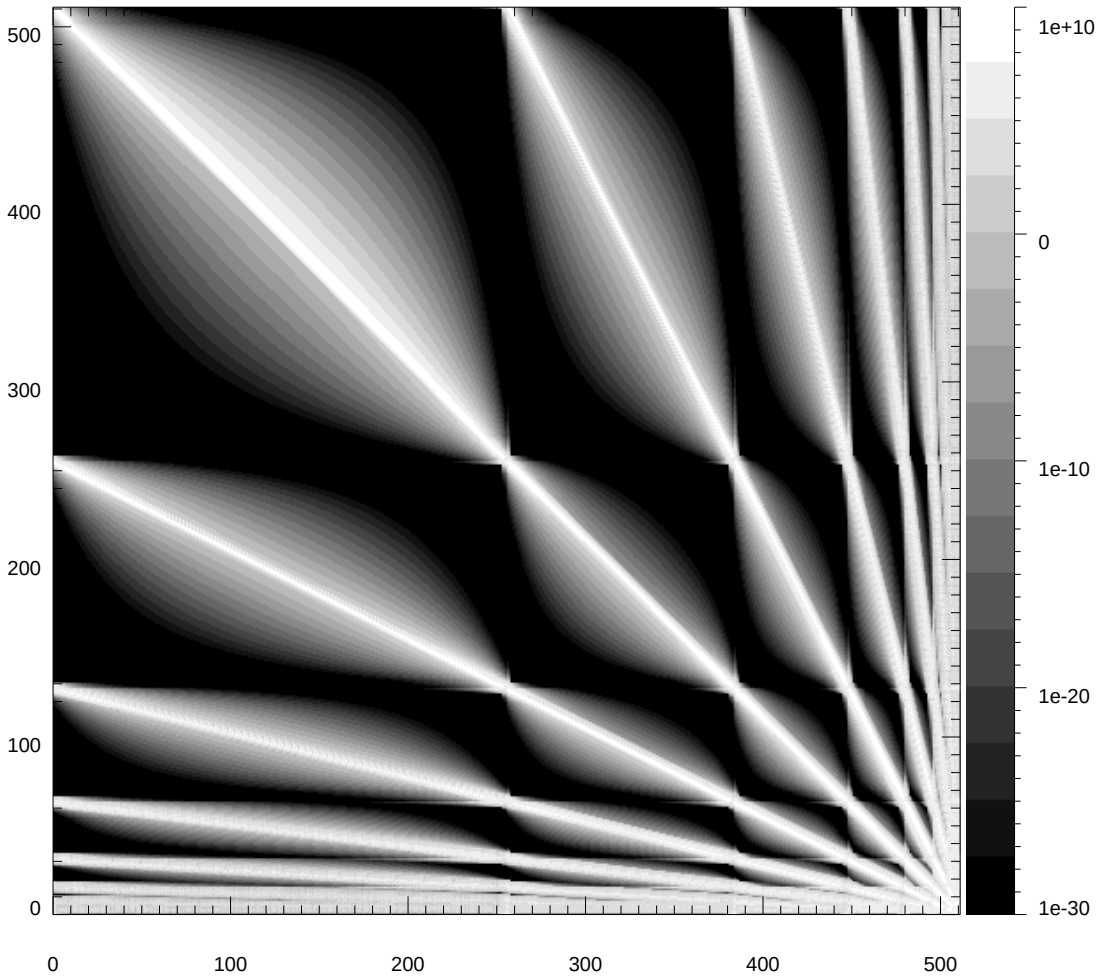


FIG. 2.3 – *Représentation de la matrice B en ondelettes (à support compact d'ordre 4) où B est la matrice de l'opérateur: $\mathcal{B} = (1 + \sin \pi x \partial_x)^{-1}$ (échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

matrice B, de telle sorte que :

$$u_p = \mathcal{B}_p f_p \tag{2.5}$$

en décomposant V_p en la somme directe de W_{p-1} et V_{p-1}, on peut écrire :

$$u_p = w_{p-1} + u_{p-1} \quad \text{et} \quad f_p = r_{p-1} + f_{p-1}$$

L'équation (2.5) peut alors se décomposer en :

$$\begin{pmatrix} w_{p-1} \\ u_{p-1} \end{pmatrix} = \begin{pmatrix} \alpha_{p-1} & \delta_{p-1} \\ \gamma_{p-1} & \beta_{p-1} \end{pmatrix} \begin{pmatrix} r_{p-1} \\ f_{p-1} \end{pmatrix}$$

avec les opérateurs suivants :

$$\alpha_{p-1} : W_{p-1} \to W_{p-1} \ , \ \delta_{p-1} : V_{p-1} \to W_{p-1} \ , \ \gamma_{p-1} : W_{p-1} \to V_{p-1} \quad \text{et} \quad \beta_{p-1} : V_{p-1} \to V_{p-1}$$

donc $\beta_{p-1} = \mathcal{B}_{p-1}$. Si les espaces W_{p-1} et V_{p-1} sont munis de, respectivement, leur base d'ondelettes et de fonctions d'échelles, la matrice associée à cette décomposition est la matrice B_1 issue de la première étape de la TOD de la matrice B :

$$B_1 = Q_p^p B \ {}^t Q_p^p$$

Q_p^p représente la matrice de la décomposition en ondelettes (section 1.1.5 équation (1.32)).

Le procédé se poursuit récursivement en posant :

$$u_{p-1} = w_{p-2} + u_{p-2} \quad \text{et} \quad f_{p-1} = r_{p-2} + f_{p-2}$$

et en décomposant $\mathcal{B}_{p-1}$ suivant les espaces W_{p-2} et V_{p-2}. Seul le terme β_{p-1} de l'étape précédente est modifié, cette modification correspondant à la deuxième étape de la TOD de la matrice B, c'est à dire la multiplication à gauche par A^{p-1}(cf section 1.1.5) et par sa transposée à droite. Le résultat du produit matrice vecteur est alors donné par :

$$u = \alpha_{p-1} r_{p-1} + \delta_{p-1} f_{p-1} + \gamma_{p-1} r_{p-1} + \alpha_{p-2} r_{p-2} + \delta_{p-2} f_{p-2} + \gamma_{p-2} r_{p-2} + \beta_{p-2} f_{p-2}$$

Pour calculer ce produit il est nécessaire de connaître non seulement les projections de f sur les espaces W_j (notées ici r_j) mais aussi les projections de f sur les espaces V_j (notées ici f_j). Et inversement on calcule des contributions au résultat u sous la forme de contributions (w_j) dans les espaces W_j, mais aussi de contributions (u_j) dans les espaces V_j.

En identifiant matrice et opérateur, cette décomposition s'écrit :[1]

$$B = \Pi_{V_p} B \Pi_{V_p} = \Pi_{V_0} B \Pi_{V_0} + \sum_{0 \le i \le p-1} (\Pi_{W_i} B \Pi_{W_i} + \Pi_{W_i} B \Pi_{V_i} + \Pi_{V_i} B \Pi_{W_i})$$

L'algorithme BCR est donc le suivant :

Étape 0: La matrice B est transformée par la TOD bidimensionnelle décrite en section 1.1.4. On obtient ainsi les blocs $\Pi_{W_i} B \Pi_{W_i} + \Pi_{W_i} B \Pi_{V_i} + \Pi_{V_i} B \Pi_{W_i}$ pour $0 \le j \le p-1$ et $\Pi_{V_0} B \Pi_{V_0}$ (voir figure 2.1).

[1] Π_V représente la projection orthogonale sur l'espace V

Étape 1: Calcul de la transformée en ondelettes de X, mais contrairement à une transformée « classique », on stocke également les coefficients de X sur les espaces V_j. On a ainsi en entrée de l'étape suivante : $\Pi_{V_i} X$ et $\Pi_{W_i} X$ pour $0 \leq i \leq p-1$.

Étape 2: Calcul des différents termes $\Pi_{W_i} B \Pi_{W_i} X$, $\Pi_{W_i} B \Pi_{V_i} X$ et $\Pi_{V_i} B \Pi_{W_i} X$, produits par blocs qui correspondent formellement au produit matrice-vecteur de la figure 2.4.

Étape 3: Le vecteur ainsi obtenu est « recompacté » en utilisant une étape de l'algorithme de synthèse en ondelettes, représenté par l'équation (1.16), à savoir :

$$\tilde{y}_{1,k} = \sum_n g_{k-2n} y_{0,n} + \sum_n h_{k-2n} z_{0,n} \tag{2.6}$$

schématisé par (avec $Y_0 = (y_{0,n})_n, Z_0 = (z_{0,n})_n, \tilde{Y}_1 = (\tilde{y}_{1,n})_n$) :

$$\begin{array}{ccc} Y_0 & \longrightarrow & \tilde{Y}_1 \\ Z_0 & \nearrow & \end{array} \tag{2.7}$$

On additionne ensuite Y_1 et $\tilde{Y}_1$ puis on itère l'étape (2.7) :

$$\begin{array}{ccc} Y_1 + \tilde{Y}_1 & \longrightarrow & \tilde{Y}_2 \\ Z_1 & \nearrow & \end{array} \tag{2.8}$$

Jusqu'à obtenir $\tilde{Y}_p = Y$.

Remarque : Cette méthode possède deux inconvénients majeurs :

- Le premier est le stockage des coefficients $\Pi_{V_j} X$ pour $j = 0, \cdots, p-1$ qui ne sont pas petits en général. On ne pourra donc pas « gagner de temps » grâce à une bonne représentation du vecteur X.

- Le deuxième est lié à l'étape 3 : le vecteur obtenu par le produit matrice-vecteur symbolisé à la figure 2.4 n'est pas l'analyse en ondelettes du vecteur Y. Il faut effectuer une opération (de coût analogue à une synthèse en ondelettes) pour obtenir Y. De même il n'est pas possible de calculer directement l'inverse ou les puissances itérées d'une matrice décomposée sous cette forme : il faut retourner dans l'espace physique pour faire ces opérations, puis redécomposer. Cette méthode est donc utilisable pour la résolution d'une série de problèmes elliptiques indépendants, mais n'est pas efficace pour résoudre une équation d'évolution où l'on doit résoudre des problèmes elliptiques successifs.

L'algorithme BCR ne permet donc pas la résolution d'une EDP (même linéaire) par une méthode basée uniquement sur la connaissance des coefficients d'ondelettes.

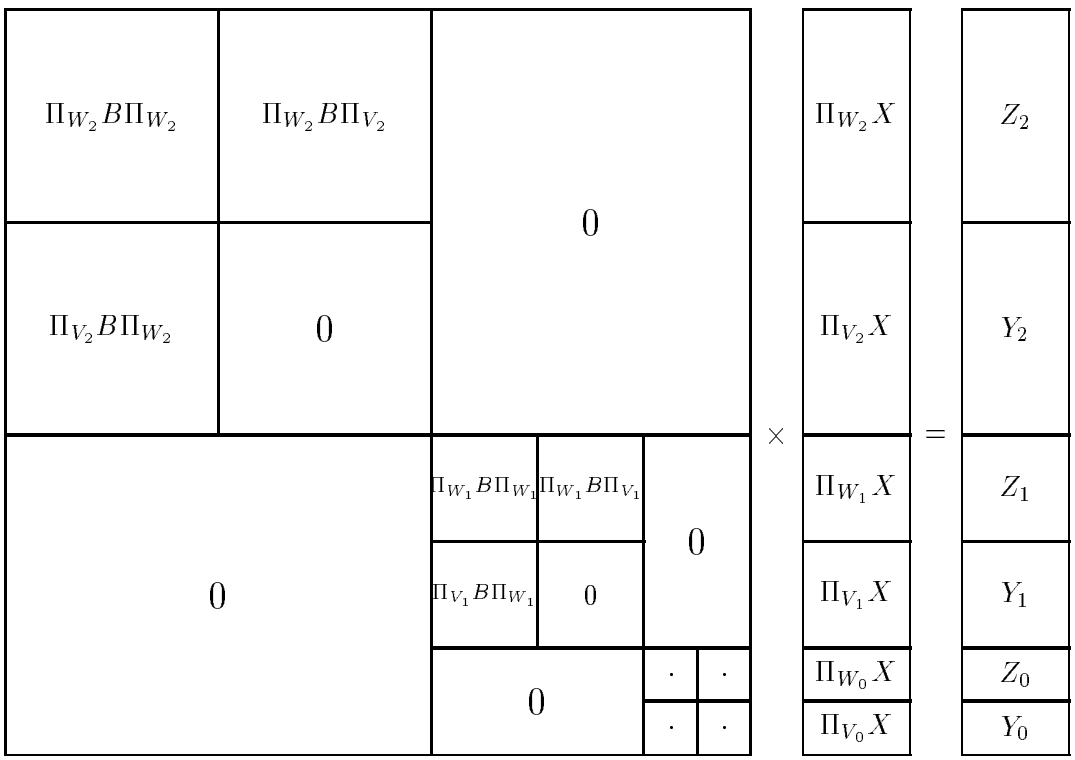


FIG. 2.4 – *Structure de la matrice B et du vecteur X décomposés par l'algorithme BCR*

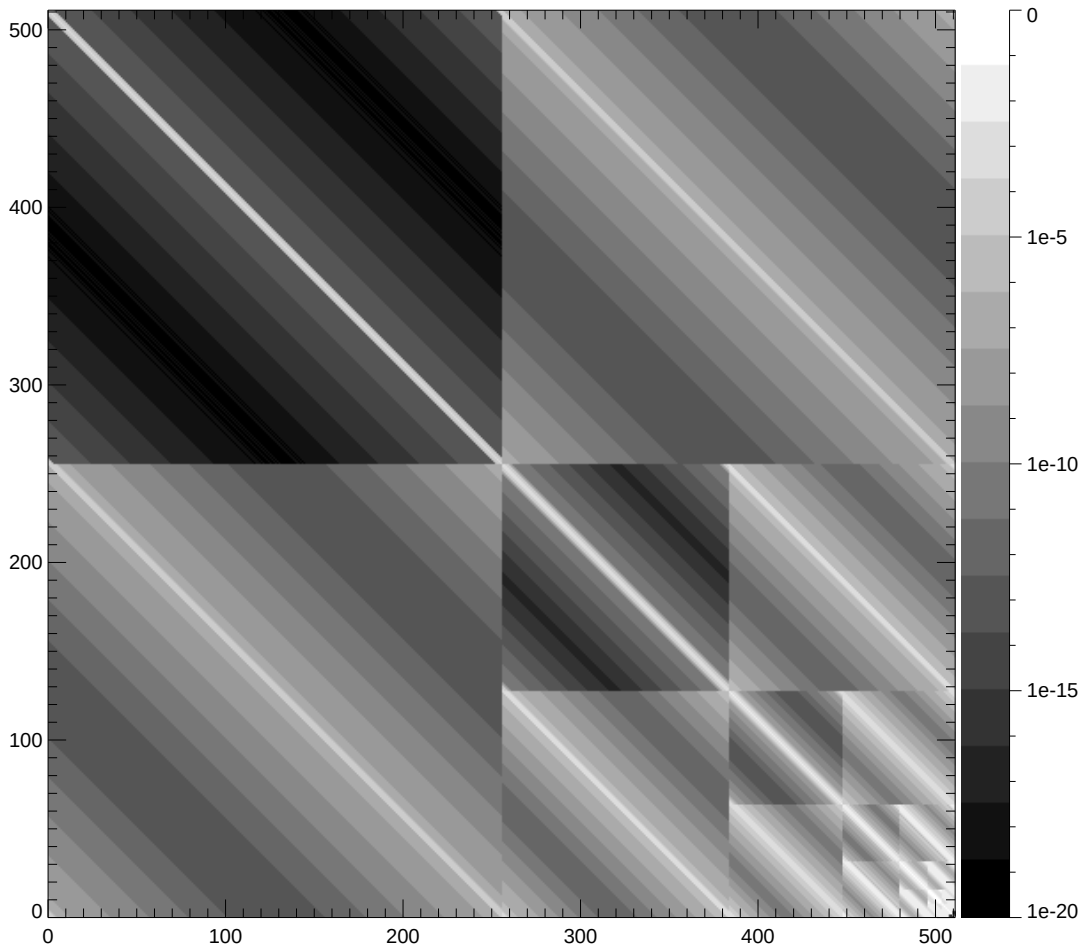


FIG. 2.5 – *Représentation BCR de la matrice B où B est la matrice de l'opérateur : $\mathcal{B} = (1 - \nu\delta t\partial^2_{xx})^{-1}$ (échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

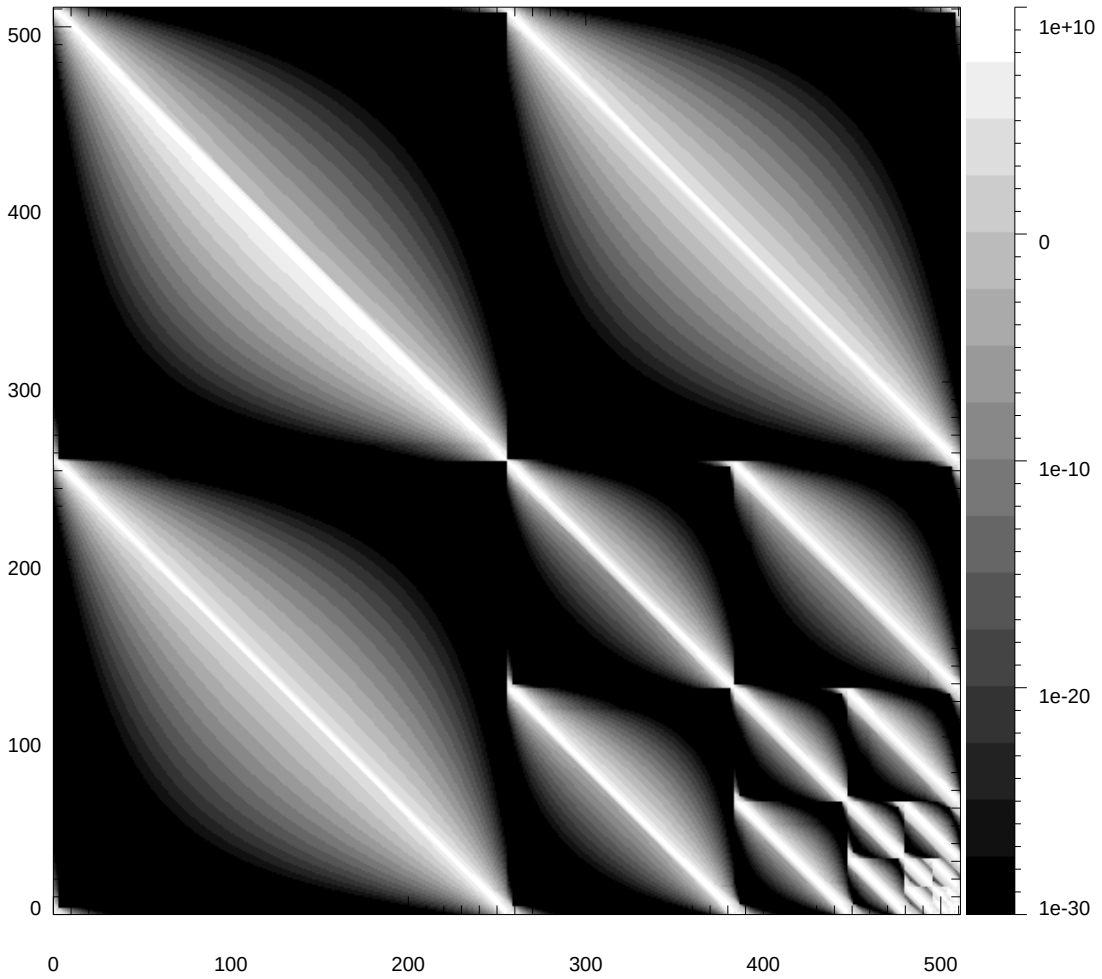


FIG. 2.6 – *Représentation BCR de la matrice B où B est la matrice de l'opérateur :* $\mathcal{B} = (1 + \sin \pi x \partial_x)^{-1}$ *(échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

2.2 Implémentation

2.2.1 Stockage Morse

La matrice B et le vecteur X sont supposés creux, on va exploiter cette propriété pour les stocker. La méthode présentée peut être employée quelle que soit la structure de la matrice.

Stockage du vecteur : Les éléments non nuls du vecteur X sont stockés dans un premier tableau VX, la position de ces éléments dans un second tableau cx, avec 0 pour marquer la fin de ce tableau.

Par exemple le vecteur $\begin{pmatrix} 2. \\ 0 \\ 5. \\ 0 \end{pmatrix}$ est stocké sous la forme :

$$\begin{array}{lllll} cx = & 1 & 3 & 0 \\ VX = & 2. & 5. \end{array}$$

Stockage de la matrice : Une technique similaire est utilisée pour la matrice B : on stocke les éléments non nuls de la matrice dans un premier tableau VB, la matrice étant parcourue colonne par colonne ; les numéros des lignes correspondantes sont stockés dans un deuxième tableau cb, enfin un troisième tableau db sert à repérer le début de chaque colonne; le premier élément de la colonne i est $VB(db(i))$. Dans ce troisième tableau, il faut également ajouter à la fin, la longueur $+1$ du tableau V afin de connaître la longueur de la dernière ligne. Par exemple, la matrice suivante :

$$B = \begin{pmatrix} 2. & 0 & 0 & 4. \\ 0 & 0 & 5. & 0 \\ 12. & 3. & 0 & 0 \\ 1. & 0 & 0 & 0 \end{pmatrix}$$

est stockée sous la forme :

$$\begin{array}{lllllllll} VB = & 2. & 12. & 1. & 3. & 5. & 4. \\ cb = & 1 & 3 & 4 & 3 & 2 & 1 & 0 \\ db = & 1 & 4 & 5 & 6 & 7 \end{array}$$

Remarque : Ce stockage n'est efficace que pour de grandes matrices creuses.

2.2.2 Produit matrice-vecteur

L'algorithme du produit $Y = BX$ est alors le suivant :

```
Y=0
```

```
i=1
   tant que cx(i) <> 0
     Pour j := db(cx(i)) jusqu'à db(cx(i)+1)-1
          Y(cb(j)) := Y(cb(j))+VB(j)*VX(i)
     i=i+1
```

Coût théorique : Le coût de l'opération est de l'ordre de $\frac{N_B N_X}{N}$ opérations (où N_B est le nombre d'éléments non nuls de B, N_X le nombre d'éléments non nuls de X et N le nombre total d'éléments de X).
Il faut ensuite comprimer le vecteur Y.

2.2.3 Produit matrice-matrice

L'algorithme précédent se généralise simplement aux produits de matrices : il suffit de considérer la deuxième matrice comme un ensemble de vecteurs. L'algorithme du produit de deux matrices B et X, $Y = BX$ est alors le suivant :[2]

```
Y=0
   Pour i := 1  jusqu'à n
      Pour j := dx(i) jusqu'à dx(i+1)-1
         Pour k:= db(cx(j)) jusqu'à db(cx(j)+1)-1
            Y(cb(k),i) := Y(cb(k),i) + VB(k) * VX(j)
```

À la section 3.3.4 nous serons amenés à calculer également le produit : $Y = X\ {}^tB$. Le stockage utilisé permet de calculer aussi ce produit de manière efficace. L'algorithme est le suivant :

```
Y=0
   Pour i := 1  jusqu'à n
      Pour j := db(i) jusqu'à db(i+1)-1
         Pour k:= dx(i) jusqu'à dx(i+1)-1
            Y(cx(k),cb(j)) := Y(cx(k),cb(j)) + VB(j) * VX(k)
```

Coût théorique : Le coût de l'opération est de l'ordre de $\frac{N_B N_X}{N}$ opérations (où N_B est le nombre d'éléments non nuls de B, N_X le nombre d'éléments non nuls de X et N le nombre total d'éléments dans une colonne).
Il faut ensuite comprimer la matrice Y.

2.3 Estimation d'erreur

Le produit matrice-vecteur $Y = BX$ ($B = (b_{i,j}), X = (x_j)$) est effectué en négligeant les termes inférieurs à ϵ_B pour la matrice B et à $\epsilon_X \|X\|$ pour le vecteur X (On note

[2] n représente la dimension des matrices (matrices carrées $N \times N$).

$\|X\| = (\sum_{i=1,N} x_i^2)^{\frac{1}{2}})$, le résultat approché est noté $\bar{Y}$. Une estimation grossière de l'erreur s'obtient par les calculs suivants :

$$\begin{aligned}\|Y-\bar{Y}\|^2 &= \sum_{|b_{i,j}|<\epsilon_B \text{ ou } |x_j|<\epsilon_X\|X\|} (\sum b_{i,j}\, x_j)^2 \leq \sum_{i=1,N} (\sum_{|b_{i,j}|<\epsilon_B \text{ ou }} b_{i,j}^2)(\sum_{|x_j|<\epsilon_X\|X\|} x_j^2) \\ &\leq \sum_{i=1,N} \left(\epsilon_B^2\|X\|^2 + \epsilon_X^2\|b_i\|^2\|X\|^2\right) \\ &\leq N\epsilon_B^2\|X\|^2 + \epsilon_X^2\|B\|^2\|X\|^2\end{aligned}$$

On a noté : $$\|b_i\| = (\sum_{j=1,N} b_{i,j}^2)^{\frac{1}{2}} \text{ et } \|B\| = (\sum_{i=1,N} (\sum_{j=1,N} b_{i,j})^2)^{\frac{1}{2}}. \quad (2.9)$$

Une majoration de l'erreur relative est donc : $$\frac{\|Y-\bar{Y}\|^2}{\|X\|^2} = O\left(N\epsilon_B^2 + \epsilon_X^2\|B\|^2\right) \quad (2.10)$$

Dans la pratique, l'erreur obtenue est beaucoup plus faible.

Des estimations d'erreur plus fines peuvent être obtenues en tenant compte de la structure particulière de la matrice B (quand B est un opérateur de Caderón-Zygmund). Ces estimations ont été établies dans [Dahmen et al.93a, Dahmen et al.94] pour des seuils ϵ_B différents suivant les blocs présents dans la matrice (voir la structure de la matrice B figures 2.1 et 2.4).

2.4 Optimisation dans le cas d'une matrice circulante

Nous présentons dans cette partie une méthode nouvelle permettant de réduire la complexité théorique et l'occupation mémoire de l'analyse en ondelettes et BCR des matrices circulantes.

Dans le cas où l'opérateur $\mathcal{L}$ du problème (3.1) est invariant par translation, la matrice A (et donc son inverse $B = A^{-1}$) est circulante. Une fois décomposée en ondelettes, ou par la méthode BCR, la matrice B est formée de blocs eux-mêmes circulants : il s'agit donc, pour chaque bloc, de ne calculer qu'une seule ligne (ou colonne), et de ne stocker sur cette ligne (ou colonne) que les éléments supérieurs à un seuil ϵ_B fixé.

Les algorithmes de décomposition de B présentés dans cette section sont fondés sur cette remarque. La réduction du coût de la décomposition et principalement du stockage est essentielle pour le passage aux dimensions supérieures. Nous exposons en premier la méthode BCR car celle-ci est plus simple.

2.4.1 Décomposition BCR

Pour décomposer la matrice B sur une base d'ondelettes 2D, il suffit, à partir des coefficients $s^j = \Pi_{V_j} B \Pi_{V_j}$ d'échelle 2^{-j} de calculer les coefficients $s^{j-1} = \Pi_{V_{j-1}} B \Pi_{V_{j-1}}$

, $\alpha^{j-1} = \Pi_{W_{j-1}} B \Pi_{W_{j-1}}$, $\beta^{j-1} = \Pi_{W_{j-1}} B \Pi_{V_{j-1}}$ et $\gamma^{j-1} = \Pi_{V_{j-1}} B \Pi_{W_{j-1}}$ d'échelle immédiatement inférieure. Ces quatre termes se calculant de manière analogue, nous ne nous intéresserons qu'au premier. On a d'après les formules (1.28–1.31):

$$s^{j-1}_{m,n} = \sum_{k=0}^{2^j-1} \sum_{l=0}^{2^j-1} h^j_k h^j_l \; s^j_{2m+k,2n+l}$$

Par hypothèse, la matrice $(s^j_{k,l})$ carrée d'ordre 2^j est circulante. Posons $s^j_{k,l} = s^j(k-l)$, avec $(s^j(n))_{n=0,2^j-1}$ la première colonne de la matrice $(s^j_{n,m})$; s^j étant périodique de période 2^j. L'équation précédente devient :

$$s^{j-1}_{m,n} = \sum_{k=0}^{2^j-1} \sum_{l=0}^{2^j-1} h^j_k h^j_l \; s^j\left(2(m-n) + (k-l)\right)$$

Sur cette formule, on constate que la matrice $(s^{j-1}_{m,n})_{m,n}$ d'ordre 2^{j-1}, est aussi circulante. On peut donc calculer uniquement sa première colonne soit $(s^{j-1}(n))_{n=0,2^{j-1}-1}$ par la formule suivante :

$$s^{j-1}(n) = s^{j-1}_{0,n} = s^j(2n) + \sum_{k=1}^{2^j-1} H^j_k \cdot \left(s^j(2n+k) + s^j(2n-k)\right) \text{ avec } H^j_k = \sum_{l=0}^{2^j-k} h^j_l h^j_{l+k} \tag{2.11}$$

qui est une convolution-décimation avec le nouveau filtre H^j.

On peut calculer aisément certains des coefficients H^j_k de ce filtre. En effet, d'après la définition des filtres (1.9-1.10), on a : $H^j_0 = \sum_{l=0}^{2^j-1} (h^j_l)^2 = 1$, ce qui explique le premier terme de la formule (2.11).
On a également (voir [Daubechies92]) :

$$H^j_{2k} = 0, \text{ pour } j = 0, \cdots, p-1 \text{ et } k = 1, \cdots, 2^{j-1} \tag{2.12}$$

Remarque :

- Avec le choix de g (1.11), on a $G^j_k = \sum_{l=0}^{2^j-k} g^j_l g^j_{l+k} = -H^j_k$
  (nouveau filtre associé au calcul des α^{j-1}, donc $\alpha^{j-1}(n) = 2s^j(2n) - s^{j-1}(n)$).
- La propriété (2.12) est conservée pour les nouveaux filtres associés au calcul des $\alpha^{j-1}, \beta^{j-1}, \gamma^{j-1}$. Par contre, les filtres associés aux β^{j-1} et γ^{j-1}, ne sont pas symétriques.

Coût théorique : Chaque étape j nécessite trois convolutions-décimations discrètes d'ordre 2^j avec des filtres de longueur L (correspondant au calcul des quatre termes $s^{j-1}, \alpha^{j-1}, \beta^{j-1}, \gamma^{j-1}$).

Le coût total de la transformation BCR de la matrice B est donc équivalent au coût de trois transformées en ondelettes unidimensionnelles, soit $6LN$ opérations pour des ondelettes à support compact, ou $9N \log_2 N$ opérations pour des ondelettes splines.

Stockage théorique : $3N - 2 = 3(\frac{N}{2} + \frac{N}{4} \cdots + 1) + 1$ coefficients si on ne tient pas compte de la lacunarité.

2.4.2 Décomposition en ondelettes

La matrice B, transformée par la méthode standard décrite à la section 2.1.2, est composée de blocs qui ne sont pas carrés (figure 2.1). On constate que ces blocs sont tout de même circulants dans le sens où pour un bloc de dimension $2^i \times 2^j$ avec $2^i \geq 2^j$ (les colonnes sont plus longues que les lignes), pour déduire une colonne de la précédente, il suffit de décaler les coefficients de 2^{i-j} lignes vers le bas. En effet, les blocs $B^{i,j} : W_j \to W_i$ (voir fig. 2.1) d'ordre $2^j \times 2^i$ vérifient : si $i \geq j$

$$B^{i,j}_{k,l} = \int_0^1 \psi^i_0(x - \frac{k}{2^i})\ \mathcal{B}\psi^j_0(x - \frac{l}{2^j})\, dx = \int_0^1 \psi^i_0(y - \frac{k - 2^{i-j}l}{2^i})\ \mathcal{B}\psi^j_0(y)\, dy = f^{i,j}_{k-2^{i-j}l} \quad (2.13)$$

en ayant posé :

$$f^{i,j}_n = \int_0^1 \psi^i_0(y - \frac{n}{2^i})\ \mathcal{B}\psi^j_0(y)\ dy$$

(si $i \leq j$ il faut permuter i et j ainsi que l et k).

Pour chaque bloc $B^{i,j}$, il va suffire de stocker la première colonne (ou la première ligne, suivant le plus grand côté du bloc), soit en tout $2N \log_2 N$ coefficients pour la matrice B entièrement stockée (N ordre de B).

Nous proposons l'algorithme de calcul suivant : notons $ss^{j,j}(n) = ss^{j,j}_{n,0}$ la première colonne de $\Pi_{V_j} B \Pi_{V_j}$, $sd^{j,j}$ la première colonne de $\Pi_{V_j} B \Pi_{W_j}$, $dd^{j,j}$ la première colonne de $\Pi_{W_j} B \Pi_{W_j}$ et $ds^{j,j}$ la première **ligne** de $\Pi_{W_j} B \Pi_{V_j}$. La suite du calcul justifiera que l'on stocke ce terme en ligne plutôt qu'en colonne.

1. La première étape est semblable à celle de la décomposition BCR : à partir de $ss^{j,j}$, on calcule : $ss^{j-1,j-1}$, $sd^{j-1,j-1}$, $dd^{j-1,j-1}$ et $ds^{j-1,j-1}$. Il faut juste prendre garde au fait que γ^{j-1} est la première colonne du bloc $\Pi_{W_j} B \Pi_{V_j}$ et $ds^{j-1,j-1}$ la première ligne, donc $ds^{j-1,j-1}(n) = \gamma^{j-1}(2^{j-1} - n)$ pour $n = 0, 2^{j-1} - 1$.

2. A la deuxième étape, on calcule : $dd^{i,j-2}$, $ds^{i,j-2}$ (stockés en lignes) et également $dd^{j-2,i}$, $sd^{j-2,i}$ (stockés en colonnes), pour $i = j-1, \cdots, p-1$. Ces quatre termes se calculent de manière analogue, détaillons le calcul de $sd^{j-2,i}$.

$$\begin{aligned} sd^{j-2,i}_{m,n} &= \sum_{l=0}^{2^{j-1}-1} g_l\ sd^{j-1,i}_{m,2n+l} \\ &= \sum_{l=0}^{2^{j-1}-1} g_l\ sd^{j-1,i}(m - 2^{i-j+1}(2n+l)) \end{aligned}$$

$$= \sum_{l=0}^{2^{j-1}-1} g_l \; sd^{j-1,i}(m - 2^{i-j+2}n - 2^{i-j+1}l)$$

On a donc :

$$sd^{j-2,i}(m) = \sum_{l=0}^{2^{j-1}-1} g_l \; sd^{j-1,i}(m - 2^{i-j+1}l) \tag{2.14}$$

On réitère ensuite l'étape 1 pour décomposer $ss^{j-1,j-1}$. Ces deux étapes sont schématisées à la figure 2.7.

Coût théorique : Le coût de la transformation est de $2L \log_2 N$ opérations pour les ondelettes à support compact ($6N \log_2^2 N$ sinon).

Stockage théorique : On doit stocker $2N \log_2 N$ éléments, sans tenir compte de la lacunarité. Dans la pratique, nous tiendrons compte également de la lacunarité des vecteurs ainsi stockés. Le calcul du produit matrice-vecteur est détaillé à la section 2.4.4.

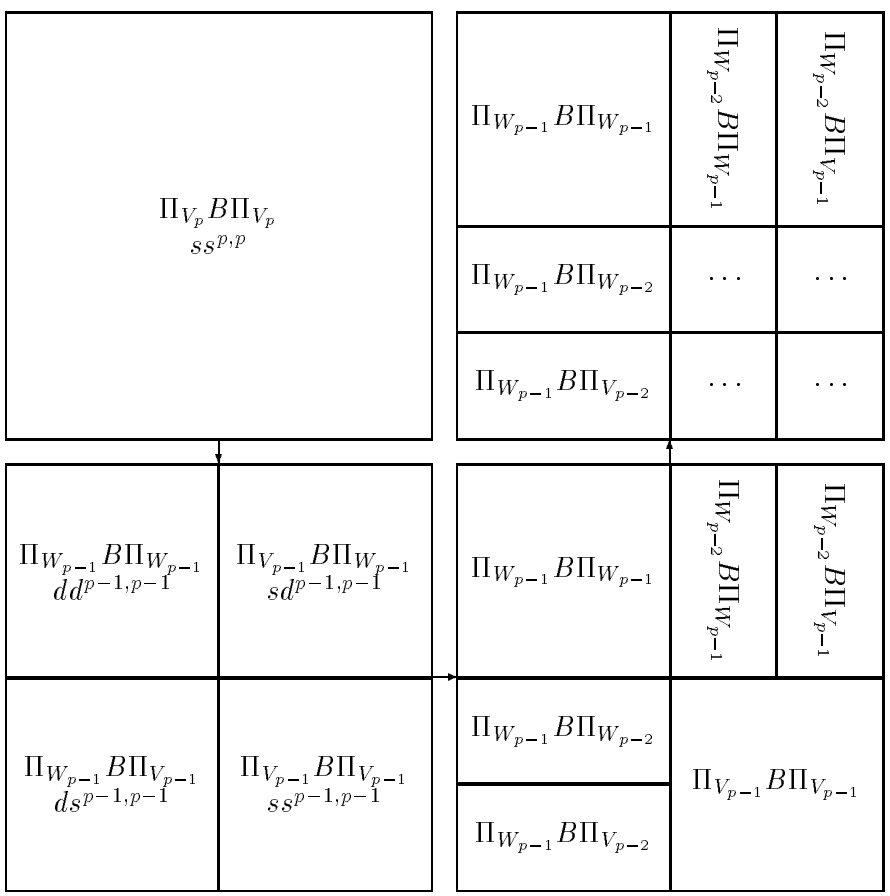


FIG. 2.7 – *Schéma de décomposition en ondelettes d'une matrice circulante*

2.4.3 Stockage

Nous allons maintenant exposer comment exploiter pratiquement les propriétés de circularité des décompositions en ondelettes des matrices circulantes.

2.4.3.1 Stockage du vecteur

Le vecteur est stocké comme précédemment, mais échelle par échelle. On a donc trois tableaux :
indv qui indique la position du premier élément de chaque échelle dans les tableau *ligv* et *valv*,
ligv contenant le numéro de ligne d'un élément non nul,
valv contenant la valeur de cet élément.

Par exemple le vecteur $\left(\begin{array}{c} 1. \\ 0. \\ 2. \\ 0. \\ \hline 0. \\ 0. \\ \hline 19. \\ \hline 21. \end{array}\right)$ est stocké sous la forme :

$$\begin{array}{rlll|l|ll|ll} indv = & 1 & & & 3 & 3 & & 4 & 5 \\ ligv = & 1 & 3 & & & 1 & & 1 & \\ valv = & 1. & 2. & & & 19. & & 21. & \end{array}$$

Ce stockage a un avantage si l'on veut utiliser un schéma adaptatif, il permet de connaître facilement les « voisins » en échelle et en espace d'un coefficient d'ondelette non nul : il peut donc être également utilisé avec des matrices non circulantes.

2.4.3.2 Matrice décomposée par la méthode standard

On a besoin de stocker seulement un vecteur par bloc de la matrice représentée en figure 2.1. Ce vecteur contient la première ligne ou colonne de ce bloc (il faut prendre l'élément le plus long des deux). On peut de plus stocker ce vecteur en tenant compte du fait qu'il est creux. La matrice suivante :

$$B = \left(\begin{array}{cccc|cc|cc} 1. & 0. & 2. & 0. & 3. & 5. & 0. & 0. \\ 0. & 1. & 0. & 2. & 0. & 0. & 8. & 0. \\ 2. & 0. & 1. & 0. & 5. & 3. & 0. & 0. \\ 0. & 2. & 0. & 1. & 0. & 0. & 10. & 14. \\ \hline 0. & 16. & 0. & 18. & 23. & 0. & 29. & 30. \\ 0. & 18. & 0. & 16. & 0. & 23. & 0. & 0. \\ \hline 19. & 0. & 0. & 22. & 25. & 0. & 33. & 34. \\ \hline 21. & 0. & 0. & 0. & 27. & 0. & 0. & 36. \end{array}\right)$$

Est stockée sous la forme :

$ind =$	1		3		5		7	8		10	11	12	13		15	16	17	18	19	20	21
$lig =$	1	3	2	4	1	4	1	1	3	1	1	1	2	4	1	1	4	1	1	1	
$val =$	1.	2.	16.	18.	19.	22.	21.	3.	5.	23.	25.	27.	8.	10.	29.	33.	14.	30.	34.	36.	

Remarque :

- Le gain d'un tel stockage est d'autant plus grand, que la matrice est grande (ici le gain en place est très faible).
- Le cas où un bloc entier est nul est traité très simplement : on trouve la même valeur deux fois de suite dans *ind*.

2.4.3.3 Matrice décomposée par la méthode non standard

Dans ce cas, on ne stocke également qu'un vecteur par bloc, et l'on a donc trois vecteurs à stocker par échelle, que l'on peut aussi stocker sous forme creuse. Le principe du stockage est le même qu'à la section précédente, mais cette fois les blocs sont tous carrés, le stockage du vecteur étant inchangé.

2.4.4 Produit matrice-vecteur

L'algorithme du produit $Y = BX$ est alors le suivant :

```
Pour chaque échelle j:=0,p-1
    Pour  cx:=indv(j),indv(j+1)-1
       l:=ligv(cx)
       x:=valv(cx)
       Pour i:=0,j-1
          Pour cy:=ind(j*p+i),ind(j*p+i+1)-1
            c:=lig(cy)
            p2:=2^(j-i)
            si (c = l) mod p2
            alors r(indr(i)+(l-c)/p2):=r(indr(i)+(l-c)/p2) + val(cy)*xl
          Fin pour cy
       Fin pour i
       Pour i:=j,p-1
          Pour cy:=ind(j*p+i),ind(j*p+i+1)-1
            c:=lig(cy)
            p2:=2^(j-i)
            r(ind(i)+c+p2*l):= r(ind(i)+c+p2*l) + val(cy)*xl
          Fin pour cy
       Fin pour i
    Fin pour cx
Fin pour j
```

Le coût théorique du produit matrice-vecteur est inchangé par rapport au cas général.

Chapitre 3

Applications

3.1 Problèmes modèles

3.1.1 Problèmes continus

Le problème modèle que nous envisageons est l'équation aux dérivées partielles, avec conditions aux limites périodiques sur $[0,1]^d$ suivante :

Trouver $u(x,t)$, $x \in [0,1]^d$, $t \in \mathbb{R}^+$ tel que

$$\begin{cases} \partial_t u + \mathcal{L}(u) = f(u) \quad \text{pour } t > 0 \\ u(x,0) = u_0(x) \\ +\text{conditions aux limites périodiques} \end{cases} \tag{3.1}$$

où $\mathcal{L}$ est un opérateur différentiel linéaire en x et f un opérateur non linéaire. Dans les exemples numériques présentés, $\mathcal{L}$ sera l'opérateur de diffusion $-\nu\Delta$, ou l'opérateur de convection $v(x)\cdot\nabla$, puis on traitera les équations de Navier-Stokes ($\mathcal{L}$ est l'opérateur de diffusion et f le terme de convection non linéaire).

Afin d'utiliser les propriétés des ondelettes indiquées dans la section 1.3.2, on considère un schéma de discrétisation en temps implicite, par exemple le schéma d'Euler rétrograde dans un souci de simplicité, pour se focaliser uniquement sur la méthode permettant d'accélérer le produit matrice-vecteur (le terme non linéaire étant traité de manière explicite) :

$$\frac{u^{n+1} - u^n}{\delta t} + \mathcal{L}(u^{n+1}) = f(u^n)$$

où δt est le pas de temps et $u^n(x) = u(x, n\delta t)$. De sorte qu'à chaque itération, on a à calculer u^{n+1} (dans un espace fonctionnel ad-hoc) tel que :

$$\mathcal{A}u^{n+1} = u^n + \delta t f(u^n) \quad \text{avec} \quad \mathcal{A} = (1 + \delta t \mathcal{L}) \tag{3.2}$$

Notre problème modèle continu sera donc : étant donnée $g \in L^2(\mathbb{T})$, trouver $u \in \mathcal{H}(\mathbb{T})$ [1] tel

[1] $\mathcal{H}(\mathbb{T})$ est un espace fonctionnel dépendant de l'opérateur $\mathcal{A}$

que :

$$\mathcal{A}u = g \quad \text{avec} \quad \mathcal{A} = (1 + \delta t \mathcal{L}). \tag{3.3}$$

On formulera aussi le problème continu sous la forme équivalente suivante : Étant donnée $g \in L^2(\mathbb{T})$, trouver $u \in \mathcal{H}(\mathbb{T})$ tel que :

$$u = \mathcal{B}g \quad \text{avec} \quad \mathcal{B}g(x) = \int_{\mathbb{T}} G(x,y)g(y)dy \tag{3.4}$$

où $G(x,y)$ est le noyau de Green de l'opérateur $\mathcal{A}$.

Nous présentons une méthode de résolution de l'EDP (3.1) basée sur des méthodes de calcul rapide de produits matrice-vecteur (chapitre 2) : après une discrétisation en espace et en temps, l'équation (3.1) est transformée en un système linéaire. Plusieurs méthodes ont ensuite été envisagées pour résoudre ce système : c'était l'objet de la section 2.1. Ces méthodes ont été optimisées au niveau du stockage dans le cas où l'opérateur est invariant par translation (section 2.4).

3.1.2 Problèmes discrets

Une discrétisation en espace ramène le problème continu (3.3) au système linéaire suivant dans un espace de dimension finie :

Étant donné $F \in \mathbb{R}^N$ trouver $U \in \mathbb{R}^N$ tel que :

$$AU = F \tag{3.5}$$

Ce système en dimension finie peut être obtenu par les démarches suivantes :

- il s'obtient par une méthode de Galerkin ou une méthode spectrale :

  On travaille dans un sous-espace E_N, de dimension finie, de $L^2(\mathbb{T})$ muni de la base $(\varphi_j)_{0 \le j < N}$. Les coordonnées de U sont les coordonnées de la projection $\Pi_{E_N} u$ de u dans cette base $u_j =< u \mid \varphi_j >$ et la matrice A de dimension $N \times N$ est définie par : $a_{ij} =< \mathcal{A}\varphi_j \mid \varphi_i >$.

- il s'obtient par une méthode de différences finies :

  Dans ce cas, les coordonnées U_i de U sont des approximations de $u(x_i)$ pour une discrétisation de $[0,1]$ et la matrice A est une approximation de différences finies de l'opérateur $\mathcal{A}$.

La formulation (3.4) devient, quant à elle, après discrétisation spatiale : Étant donné $F \in \mathbb{R}^N$ trouver $U \in \mathbb{R}^N$ tel que :

$$U = BF \tag{3.6}$$

La matrice B étant cette fois définie par :

- pour une méthode de Galerkin :

  $b_{ij} =< \mathcal{B}\varphi_j \mid \varphi_i >= \int\int G(x,y)\varphi_i(x)\varphi_j(y)\ dxdy$

– pour une méthode de différences finies :

b_{ij} est une approximation de $G(x_i, y_j) = \int\int G(x,y)\delta_{x_i}(x)\delta_{y_j}(y)\ dxdy$.

D'autres approches, liées aux méthodes de Petrov-Galerkin et aux bases biorthogonales d'ondelettes permettent la discrétisation de (3.3) et sont développées dans [Liandrat et al.91, Dahlke et al.94]. Ces méthodes englobent également celles présentées ci-dessus. Il s'agit en fait d'écrire f sous la forme $f = \sum_i < f \mid \varphi_i > \tilde{\varphi}_i$ et u sous la forme $u = \sum_i < u \mid \theta_i > \tilde{\theta}_i$, θ étant une fonction dérivée de l'ondelette φ.

3.1.3 Résolution du système linéaire

Pour des discrétisations type différences finies, A est une matrice creuse (par exemple matrice tri-diagonale dans le cas de méthodes d'ordre 2). Le système (3.5) se résout classiquement par une méthode directe, avec un coût de l'ordre de $O(N)$ opérations. Pour une dimension d'espace supérieure ou égale à deux, on utilise plutôt une méthode itérative, (la largeur de bande de la matrice étant dans ce cas N pour un problème de dimension N^2, et cette bande se remplit complètement avec une méthode directe).

L'originalité de ce travail réside dans le fait que nous utilisons la formulation (3.6), c'est à dire qu'on suppose que la matrice A (invariante au cours du temps) a été préalablement inversée et la résolution du système linéaire est ramenée à un simple produit matrice-vecteur. L'inversion de la matrice utilise une méthode classique (voir annexe A.1) et est effectuée une seule fois au début du calcul. Contrairement à la matrice A, $B = A^{-1}$ est une matrice pleine (discrétisation d'un noyau de Green). A priori, le produit matrice vecteur a une complexité de l'ordre de N^2 opérations si N est l'ordre de la matrice, donc largement supérieure aux méthodes signalées plus haut. Mais B étant la discrétisation d'un noyau de Green, elle sera presque diagonale sur une base d'ondelettes (section 1.3.1). Notre but est donc, en utilisant cette propriété, de transformer B pour la rendre creuse. Les méthodes présentées au chapitre 2 seront envisagées et comparées, l'objectif étant de trouver la méthode qui minimisera le coût du produit matrice-vecteur, celui-ci étant fonction de la lacunarité de la matrice B et de celle du vecteur U, ainsi que le coût du stockage de la matrice B.

3.2 Expériences numériques en dimension un

Nous présentons dans cette partie une comparaison des différentes méthodes pour la résolution numérique du problème (3.1) en dimension un, sans opérateur non linéaire, successivement pour l'opérateur de diffusion et pour l'opérateur de convection. Les temps de calculs présentés dans ce chapitre correspondent à des tests numériques effectués sur HP 720.

3.2.1 Présentation des problèmes tests

3.2.1.1 Équation de diffusion

Le premier problème test considéré est l'équation de la chaleur avec conditions aux limites périodiques.

$$\begin{cases} \frac{\partial}{\partial t}u - \nu \frac{\partial^2}{\partial x^2}u = 0 \quad x \in [0,1] \, , t \geq 0 \\ u(x,0) = u_0^1(x) = \begin{cases} x & \text{si } x \in [0, \frac{1}{2}[\\ x-1 & \text{si } x \in [\frac{1}{2}, 1[\end{cases} \end{cases} \tag{3.7}$$

Après discrétisation en temps (schéma d'Euler implicite) et en espace (différences finies)[2], (3.7) s'écrit :

$$U^{n+1} = B_1 U^n \tag{3.8}$$

B_1 étant l'inverse de la matrice : ($\nu \Delta t$ pris égal à 1)

$$B_1^{-1} = \begin{pmatrix} 3 & -1 & 0 & \cdots & 0 & 0 & -1 \\ -1 & 3 & -1 & \cdots & 0 & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & -1 & 3 & -1 \\ -1 & 0 & 0 & \cdots & 0 & -1 & 3 \end{pmatrix}$$

B_1 est une matrice pleine, comme le montre la figure 3.1.

3.2.1.2 Équation de transport

Le second problème test est l'équation de transport avec conditions aux limites périodiques :

$$\begin{cases} \frac{\partial}{\partial t}u + \sin(\pi x) \frac{\partial}{\partial x}u = 0 \quad , x \in [0,1] \, , \; t \geq 0 \\ u(x,0) = u_0^2(x) \end{cases} \tag{3.9}$$

$u_0^2(x)$ sera une coupe d'un champ turbulent bidimensionnel (obtenu par simulation numérique [Basdevant et al.81]). Les mêmes discrétisations en temps et en espace qu'au paragraphe précédent conduisent au système :

$$U^{n+1} = B_2 U^n \tag{3.10}$$

B_2 est représenté à la figure 3.2.

[2] On peut également utiliser une méthode de Galerkin, mais une analyse spectrale montre que ces deux types d'approches ont les mêmes propriétés si l'on utilise comme base les ondelettes à support compact de Daubechies (elles sont d'ailleurs identiques pour l'ordre 2, pour les ordres supérieurs la méthode de Galerkin génère des schémas aux différences finies centrés non optimaux dans le sens où, pour un ordre donné, le schéma utilise un nombre de points supérieur au schéma aux différences finies de même ordre, voir [Beylkin92, Jameson94, Walden])

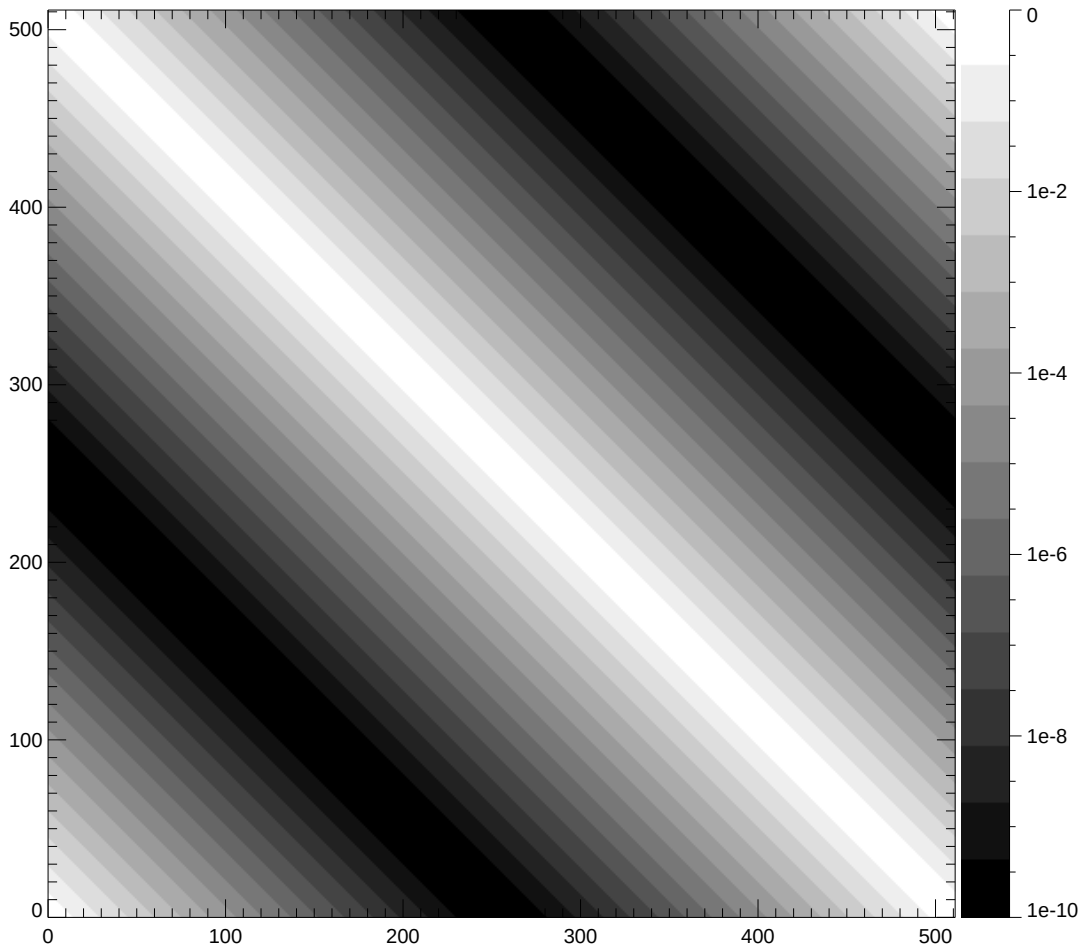


FIG. 3.1 – *Représentation matricielle de l'opérateur* $(I - \Delta)^{-1}$ *(échelle logarithmique)*

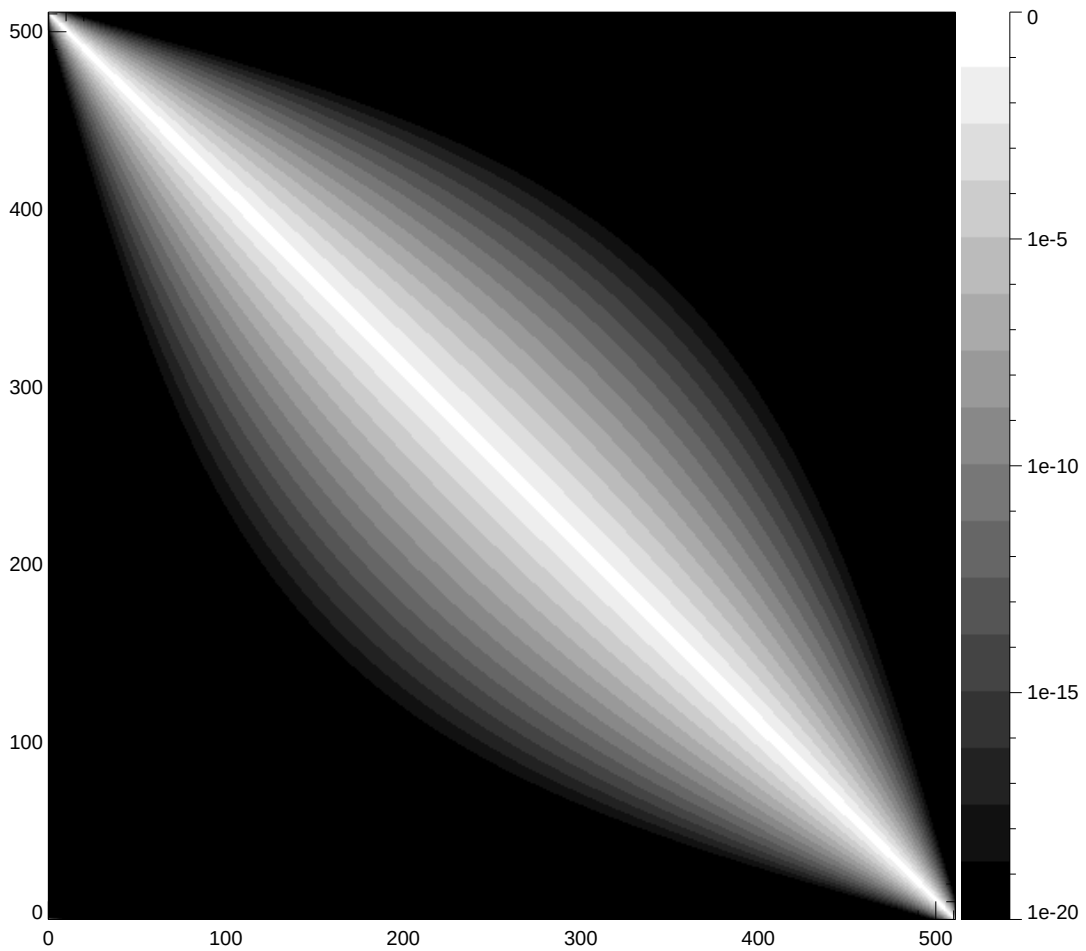


FIG. 3.2 – *Représentation matricielle de l'opérateur* $(I - \sin(\pi x)\frac{\partial}{\partial x})^{-1}$ *(échelle logarithmique)*

3.2.2 Ajustement des paramètres

- Les systèmes (3.8) et (3.10) sont décomposés d'une part en ondelettes (forme standard), d'autre part par l'algorithme BCR (forme non standard).

- Les ondelettes utilisées pour les simulations sont des ondelettes à support compact [Daubechies92] d'un ordre r variable. Rappelons que l'ondelette mère ψ d'ordre r a pour support l'intervalle $[0, 2r-1]$ et est orthogonale aux polynômes de degré $\leq r$. Le choix des ondelettes à support compact a été motivé par la rapidité des algorithmes de décomposition-recomposition (comparativement aux autres bases d'ondelettes) : une conséquence de ce choix est de moins pénaliser la méthode BCR, qui utilise une décomposition-recomposition à chaque étape de résolution.

- Un seuil ϵ_B est appliqué sur les coefficients des matrices B_1 et B_2, un seuil relatif différent ϵ_U est fixé sur les vecteurs U^n : en dessous de $\epsilon_U \|U^n\|$ les coefficients sont mis à 0.

- Nous faisons ensuite varier différents paramètres : nombre de points de discrétisation (i.e. l'ordre des matrices B_1 et B_2), ordre de l'ondelette, seuils sur la matrice et les vecteurs, afin de comparer les temps de calcul pour les deux méthodes appliquées aux équations (3.7) et (3.9).

3.2.3 Comparaison des lacunarités

3.2.3.1 En fonction de l'ordre de la matrice

Les figures (3.3–3.4) représentent respectivement les taux de compression des opérateurs B_1 et B_2. Sur chaque figure on compare les taux de compression obtenus par la décomposition BCR (forme non standard) et par la décomposition en ondelettes (forme standard) en fonction de l'ordre N de la matrice. Le taux de compression est donné par le rapport du nombre de coefficients de B_1 (ou B_2) supérieurs à un seuil $\epsilon_B = 10^{-5}$, sur le nombre total N^2 de coefficients de B_1 (ou B_2).

On constate sur ces figures que les deux méthodes conduisent à des taux de compression très voisins : les courbes suivent une loi en $O(\frac{\log_2 N}{N})$ où N est l'ordre de la matrice. Cette loi s'explique aisément : si l'on suppose que chaque bloc dans la matrice B_1 (ou B_2), écrite sous forme standard ou non standard, est diagonale-bande quand on annule ses coefficients inférieurs à ϵ_B (ce qui est le cas pour les opérateurs de Calderón-Zygmund, voir [Beylkin et al.91]), la lacunarité de la matrice B_1 sera proportionnelle au nombre de blocs de cette matrice (soit $O(N \log_2 N)$ pour la représentation standard et $O(N)$ pour la représentation BCR) divisé par le nombre total de coefficients (soit N^2).

3.2.3.2 En fonction de l'ordre de l'ondelette

Les figures 3.5 et 3.6 donnent les taux de compression des opérateurs B_1 et B_2 en fonction de l'ordre r de la base d'ondelettes. L'ordre de B_1 et B_2 est choisi égal à $N = 1024$,

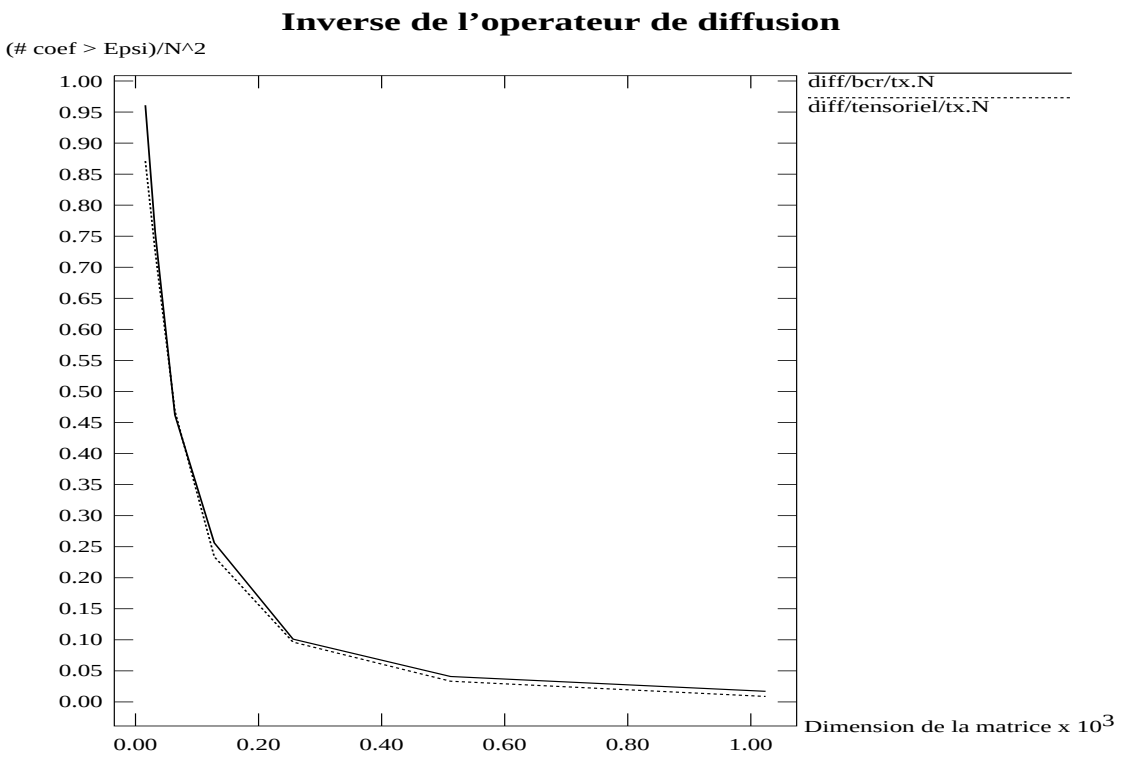


FIG. 3.3 – *Taux de compression de l'opérateur B_1 en fonction de l'ordre de la matrice (seuil $\epsilon_B = 10^{-5}$, ondelettes d'ordre 4).*
1) Trait plein: forme non standard (BCR) 2) Pointillé: forme standard

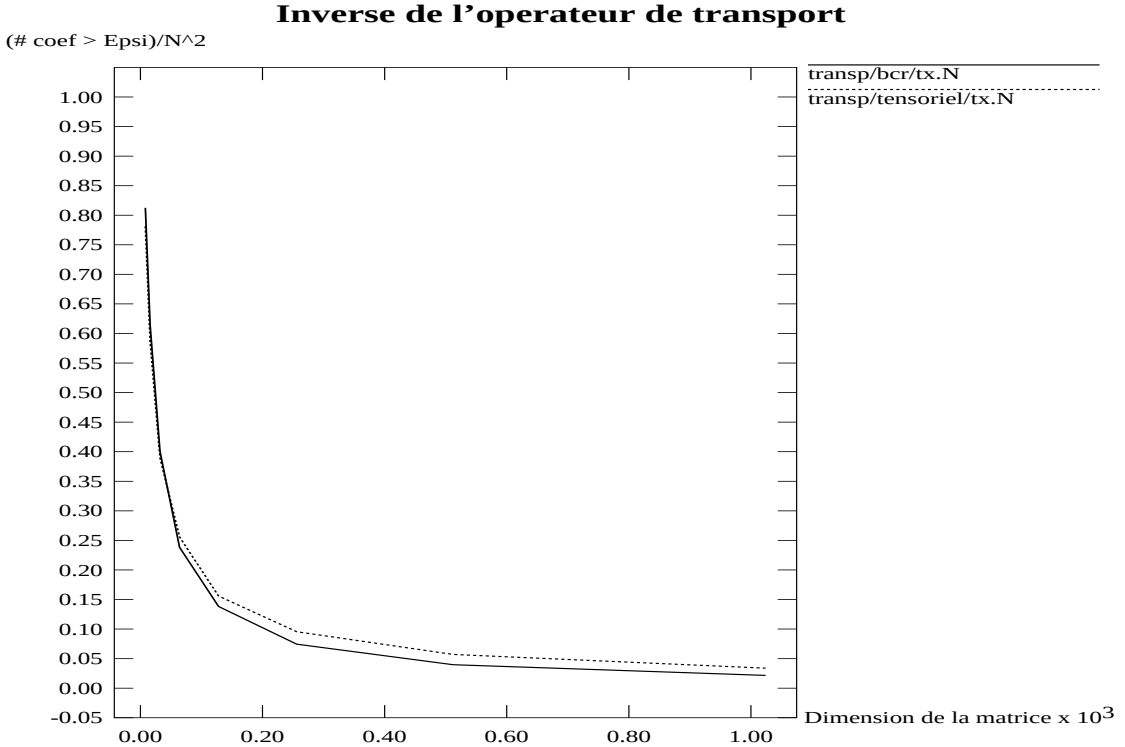


FIG. 3.4 – *Taux de compression de l'opérateur B_2 en fonction de l'ordre de la matrice (seuil $\epsilon_B = 10^{-5}$, ondelettes d'ordre 4).*
1) Trait plein: forme non standard (BCR) 2) Pointillé: forme standard

le seuil $\epsilon_B = 10^{-5}$. Plus r est grand, plus les ondelettes sont régulières, ont un support large dans l'espace physique et concentré dans l'espace de Fourier. L'ordre un (base de Haar) donne une mesure aberrante par rapport aux ordres supérieurs, pour lesquels les courbes sont constantes : la diminution du support dans l'espace de Fourier est exactement compensée par l'augmentation du support de l'ondelette dans l'espace physique. Le choix de l'ordre de l'ondelette se fera donc essentiellement en fonction du taux de compression obtenu sur le vecteur, celui-ci étant lié à la régularité de la solution de l'équation.

3.2.3.3 En fonction du seuil sur la matrice

Les figures (3.7–3.8) donnent les taux de compression des opérateurs B_1 et B_2 en fonction du seuil ϵ_B. Les ondelettes ont été choisies d'ordre $r = 4$, $N = 1024$, ces courbes variant très peu en fonction de r (voir figures 3.5 et 3.6).

Dans tous les cas, les courbes suivent approximativement un comportement linéaire en ϵ_B.

3.2.4 Erreur et temps de calcul en fonctions des seuils sur l'opérateur et la solution

Dans cette partie nous calculons l'erreur avec la solution de référence Y_1 (ou Y_2) en fonction du seuil ϵ_B imposé sur l'opérateur B_1 (ou B_2) et du seuil ϵ_U imposé sur la solution U^n à chaque pas de temps. Les solutions de référence ont été calculées en utilisant la méthode standard avec pour seuils : $\epsilon_B = \epsilon_U = 10^{-15}$ (figures 3.9-3.10).

3.2.4.1 Équation de la chaleur par la méthode standard

Dans le tableau 3.11, nous avons représenté, en fonction de ϵ_B et ϵ_U, les lacunarités respectives de l'opérateur B_1 décomposé en ondelettes, de la condition initiale U_0 et de la solution U_F de l'équation (3.7) pour $t = 0.1$. En avant-dernière colonne figure l'erreur (en norme l^2) avec la solution de référence Y_1. L'ordre de la matrice B_1 est choisi égal à $N = 1024$, les ondelettes sont d'ordre 4, la viscosité $\nu = 1$ et $\delta t = 10^{-3}$ (on a donc effectué 100 itérations en temps).

Malgré le nombre relativement élevé d'itérations, l'erreur reste faible par rapport aux seuils, et pour cette équation, il faut choisir un seuil plus petit sur la matrice que sur le vecteur pour minimiser l'erreur.

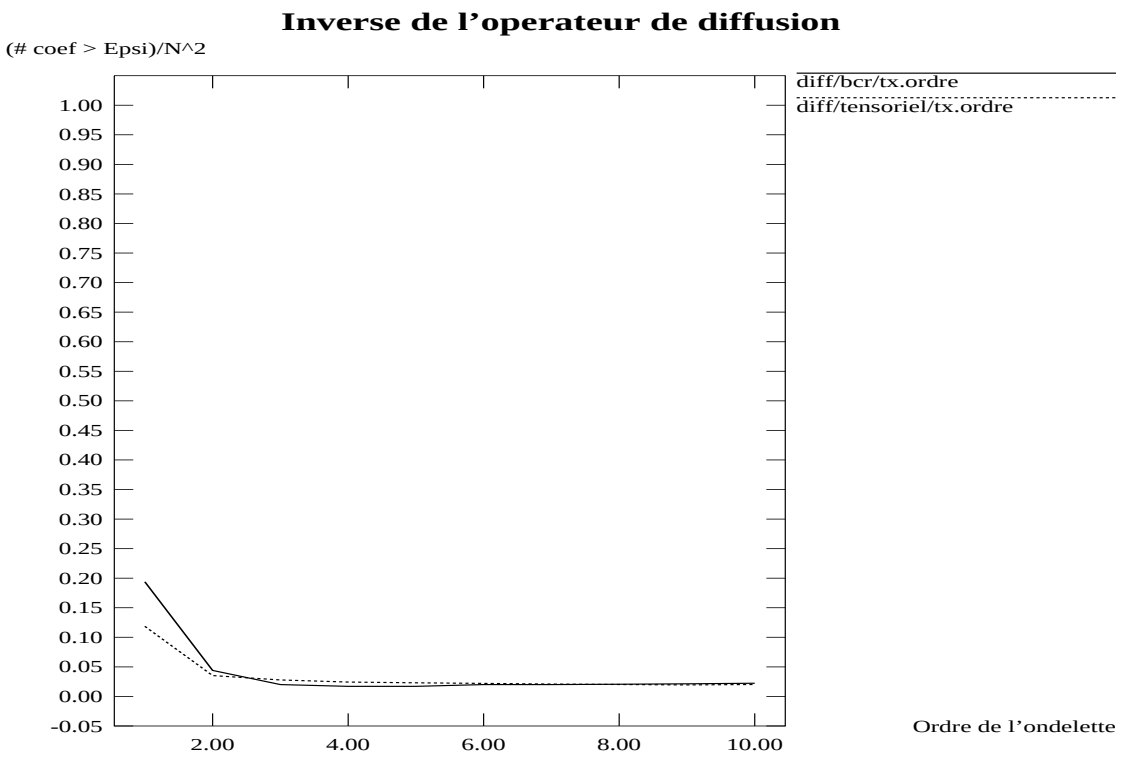


FIG. 3.5 – *Taux de compression de l'opérateur B_1 en fonction de l'ordre de l'ondelette de base (seuil $\epsilon_B = 10^{-5}$, N=1024)*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

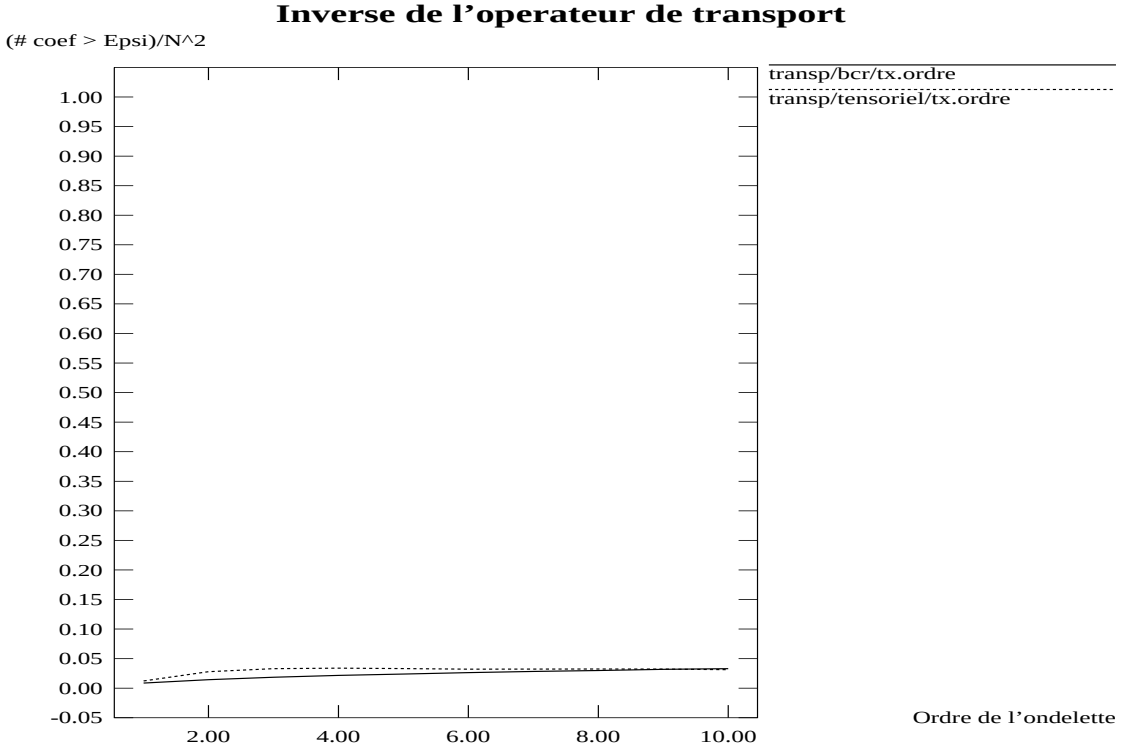


FIG. 3.6 – *Taux de compression de l'opérateur B_2 en fonction de l'ordre de l'ondelette de base (seuil $\epsilon_B = 10^{-5}$, N=1024)*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

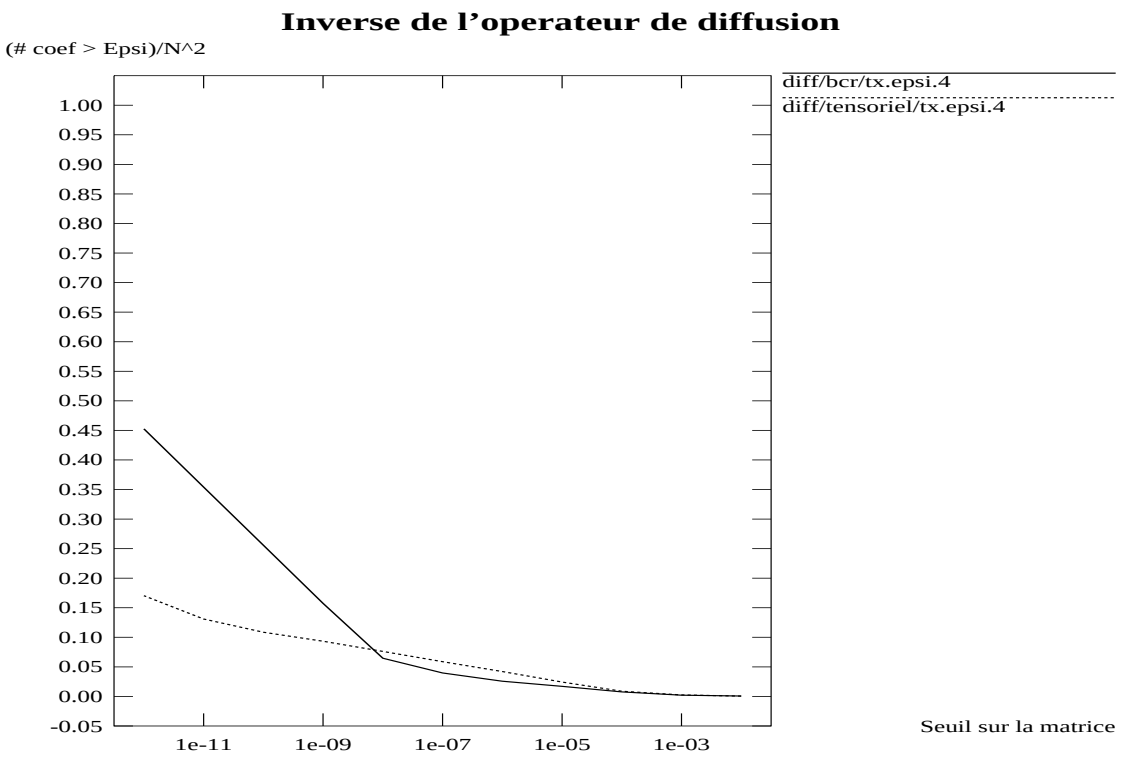


FIG. 3.7 – *Taux de compression de l'opérateur B_1 en fonction du seuil sur la matrice ϵ_B (N=1024)*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

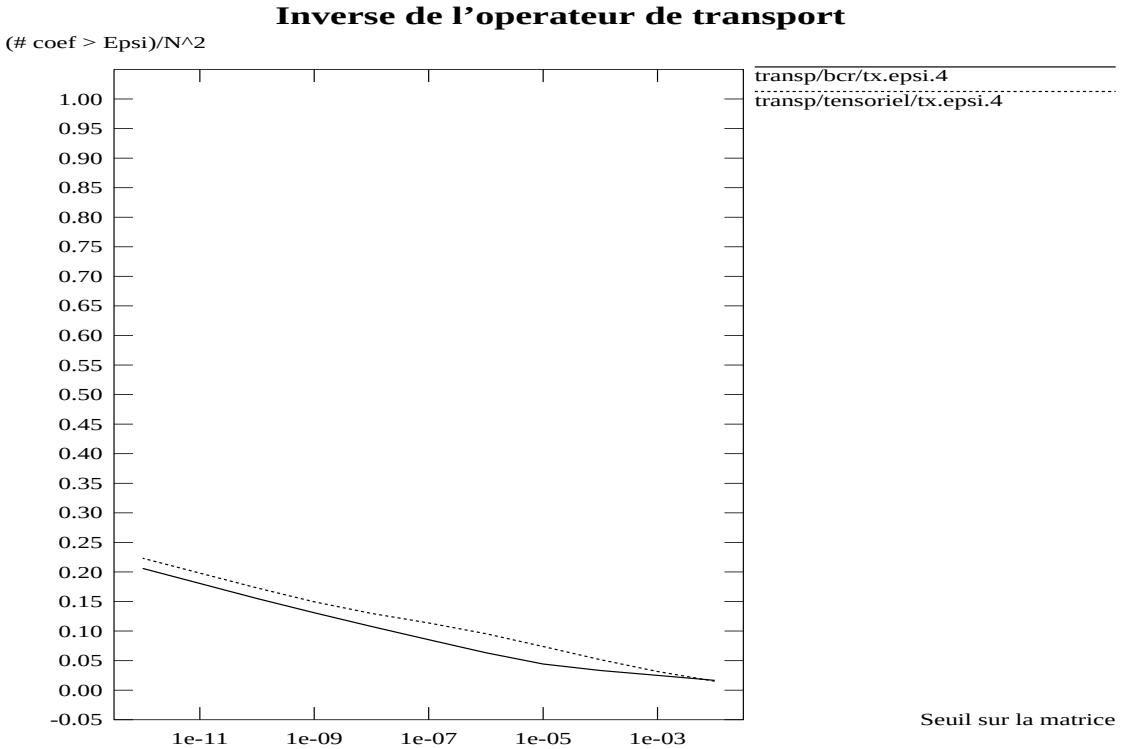


FIG. 3.8 – *Taux de compression de l'opérateur B_2 en fonction du seuil sur la matrice ϵ_B (N=1024)*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

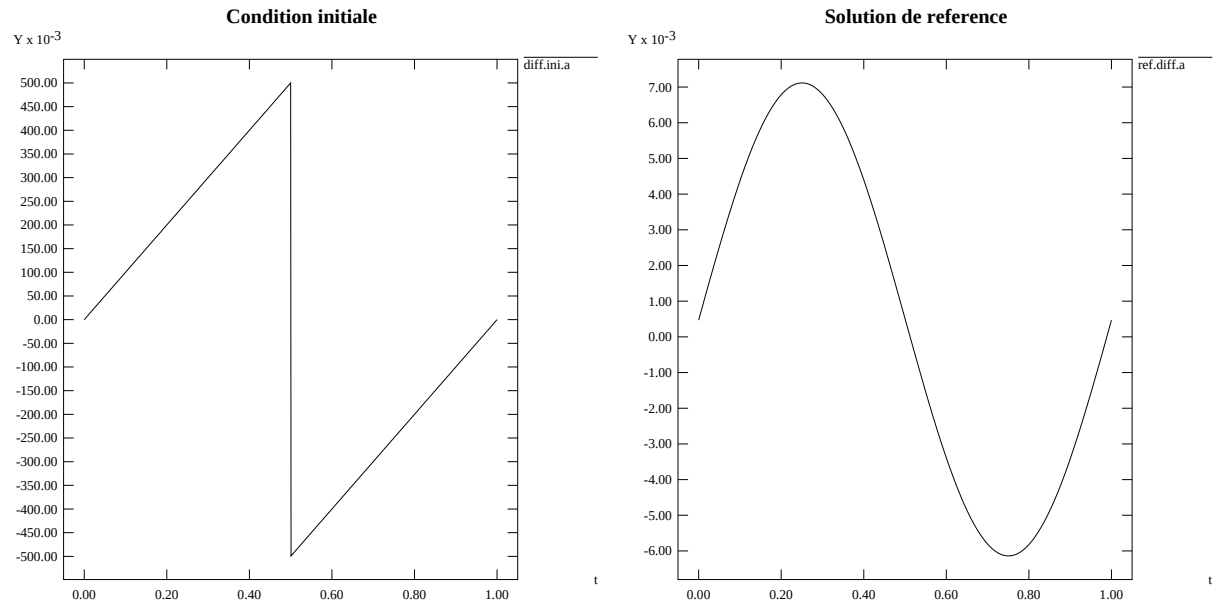


FIG. 3.9 – *Condition initiale et solution de référence au temps* $t = 0,1$ *(Équation de diffusion)*

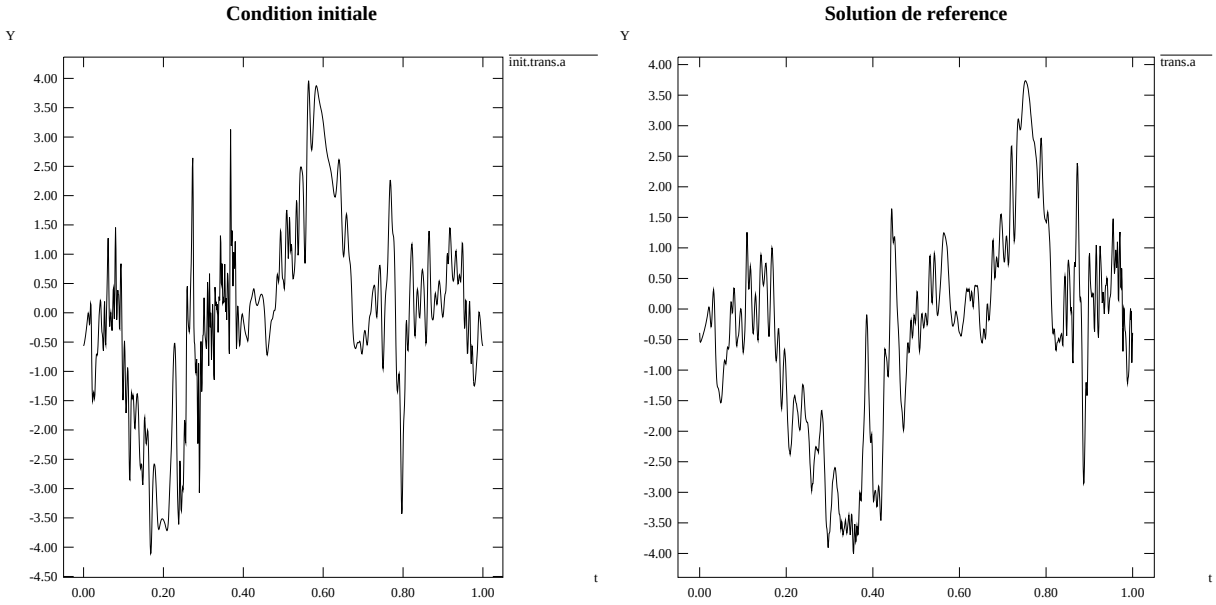


FIG. 3.10 – *Condition initiale et solution de référence au temps* $t = 0,2$ *(Équation de transport)*

ϵ_B	ϵ_U	# coef $B_1 > \epsilon_B$	# coef $U_0 > \epsilon_U \|U_0\|$	# coef $U_F > \epsilon_U \|U_F\|$	$\frac{\|U_F - Y_1\|_2}{\|U_F\|_2}$	temps (s)
10^{-2}	10^{-2}	579	30	51	$1.1 \cdot 10^{-4}$	0.06
10^{-2}	10^{-3}	579	42	64	$2.9 \cdot 10^{-5}$	0.06
10^{-3}	10^{-2}	2659	30	53	$1.0 \cdot 10^{-4}$	0.07
10^{-3}	10^{-3}	2659	40	84	$1.3 \cdot 10^{-6}$	0.08
10^{-4}	10^{-2}	9259	30	58	$1.0 \cdot 10^{-4}$	0.10
10^{-4}	10^{-3}	9259	40	100	$1.7 \cdot 10^{-7}$	0.14
10^{-5}	10^{-2}	25559	30	31	$1.0 \cdot 10^{-4}$	0.11
10^{-5}	10^{-3}	25559	40	100	$3.5 \cdot 10^{-8}$	0.25
10^{-5}	10^{-4}	25559	42	200	$3.5 \cdot 10^{-8}$	0.40
10^{-8}	10^{-2}	79851	30	31	$1.0 \cdot 10^{-4}$	0.51
10^{-8}	10^{-3}	79851	40	60	$5.0 \cdot 10^{-10}$	0.76
10^{-8}	10^{-4}	79851	42	116	$4.9 \cdot 10^{-10}$	1.09
10^{-8}	10^{-5}	79851	47	208	$5.4 \cdot 10^{-11}$	1.43
10^{-8}	10^{-6}	79851	50	494	$5.4 \cdot 10^{-11}$	2.08

FIG. 3.11 – *Comparaison de l'erreur avec la solution de référence et temps de calcul (100 itérations) en fonction des seuils sur l'opérateur et la solution (équation de la chaleur par la méthode standard)*

Un bon compromis est atteint avec $\epsilon_B = 10^{-2}$, $\epsilon_U = 10^{-3}$. La solution U_F de (3.7) est alors représentée avec 64 coefficients d'ondelettes dans un espace de dimension $N = 1024$, pour une erreur l^2 relative de l'ordre de 10^{-5}, et un temps de calcul très faible (0.06s).

Pour un ϵ_B donné, il est inutile d'utiliser un ϵ_U inférieur aux valeurs du tableau car ensuite l'erreur stagne : on ne fait qu'augmenter le nombre de degrés de liberté au détriment du temps de calcul (voir les données concernant $\epsilon_B = 10^{-5}$ et $\epsilon_B = 10^{-8}$).

3.2.4.2 Équation de la chaleur par la méthode BCR

Rappelons que pour la méthode BCR, il faut stocker à la fois les coefficients d'ondelettes et les coefficients d'échelles de la solution, les colonnes # coef U_0 et # coef U_F du tableau suivant représentent donc la somme des coefficients d'ondelettes et des coefficients d'échelles.

Les choix les plus avantageux pour les seuils ϵ_B et ϵ_U sont ceux correspondants à ϵ_B et ϵ_U du même ordre. Il faut noter que la complexité est très élevée, ceci est dû à la présence des coefficients de la solution sur les fonctions d'échelles. Les temps de calculs sont nettement plus importants que pour la méthode standard (tableau 3.11) à cause d'une part, du plus grand nombre de coefficients et d'autre part, de l'obligation d'effectuer un retour dans l'espace physique.

Les meilleurs compromis sont obtenus avec $\epsilon_B = 10^{-2}$ et $\epsilon_U = 10^{-2}$ ou $\epsilon_B = 10^{-5}$ et $\epsilon_U = 10^{-4}$.

ϵ_B	ϵ_U	# coef $B_1 > \epsilon_B$	# coef $U_0 > \epsilon_U \|U_0\|$	# coef $U_F > \epsilon_U \|U_F\|$	$\frac{\|U_F - Y_1\|_2}{\|U_F\|_2}$	temps (s)
10^{-2}	10^{-2}	718	734	1084	$9.5 \cdot 10^{-5}$	1.28
10^{-2}	10^{-4}	718	1057	1086	$1.3 \cdot 10^{-4}$	1.30
10^{-3}	10^{-2}	2286	734	1144	$9.7 \cdot 10^{-5}$	1.34
10^{-3}	10^{-3}	2286	1024	1150	$9.7 \cdot 10^{-5}$	1.47
10^{-4}	10^{-2}	7974	734	1099	$1.0 \cdot 10^{-4}$	1.57
10^{-4}	10^{-3}	7974	1024	1426	$1.0 \cdot 10^{-4}$	1.96
10^{-5}	10^{-2}	17894	734	1053	$1.0 \cdot 10^{-4}$	1.61
10^{-5}	10^{-3}	17894	1024	1231	$1.0 \cdot 10^{-4}$	2.29
10^{-5}	10^{-4}	17894	1057	1682	$5.7 \cdot 10^{-8}$	2.64
10^{-8}	10^{-2}	67798	734	1053	$1.0 \cdot 10^{-4}$	4.17
10^{-8}	10^{-3}	67798	1024	1081	$1.0 \cdot 10^{-4}$	4.31
10^{-8}	10^{-4}	67798	1057	1134	$5.6 \cdot 10^{-10}$	4.61
10^{-8}	10^{-5}	67798	1067	1264	$2.8 \cdot 10^{-10}$	5.12
10^{-8}	10^{-6}	67798	1070	1979	$2.8 \cdot 10^{-10}$	7.57

FIG. 3.12 – *Comparaison de l'erreur avec la solution de référence et temps de calcul (100 itérations) en fonction des seuils sur l'opérateur et la solution (équation de la chaleur méthode BCR)*

3.2.4.3 Équation de transport par la méthode standard

Ce tableau 3.13 est analogue au tableau 3.11, l'opérateur B_1 étant remplacé par l'opérateur B_2 décomposé en ondelettes. Sur ce tableau , on constate que l'opérateur de transport est moins creux que l'opérateur de diffusion. Les meilleurs compromis sont obtenus pour ϵ_B du même ordre que ϵ_U. La solution étant très chahutée, beaucoup de coefficients d'ondelettes de petites échelles sont significatifs : c'est pourquoi si le seuil ϵ_U est petit, la diminution du nombre de degrés de liberté est très faible (voir nulle pour $\epsilon_U = 10^{-6}$), cette méthode est donc intéressante pour une précision limitée ($\sim 10^{-3}$) sur la solution.

3.2.4.4 Équation de transport par la méthode BCR

Ce tableau 3.14 est analogue au tableau 3.12, l'opérateur B_1 étant remplacé par l'opérateur B_2 décomposé par la méthode non standard. On constate le même phénomène qu'au tableau 3.12 en ce qui concerne les taux de compression et le temps de calcul, beaucoup plus élevés que pour la méthode standard. Le nombre de coefficients est toujours supérieur au nombre de composantes du vecteur U dans l'espace physique.

ϵ_B	ϵ_U	# coef $B_2 > \epsilon_B$	# coef $U_0 > \epsilon_U \|U_0\|$	# coef $U_F > \epsilon_U \|U_F\|$	$\frac{\|U_F - Y_1\|_2}{\|U_F\|_2}$	temps (s)
10^{-2}	10^{-2}	1025	174	174	$1.3 \cdot 10^{-3}$	6.75
10^{-2}	10^{-3}	1025	516	516	$1.3 \cdot 10^{-3}$	12.62
10^{-3}	10^{-2}	7141	174	899	$1.2 \cdot 10^{-3}$	30.97
10^{-3}	10^{-3}	7141	516	981	$1.2 \cdot 10^{-3}$	32.96
10^{-4}	10^{-2}	18996	174	879	$1.1 \cdot 10^{-3}$	63.59
10^{-4}	10^{-3}	18996	516	1011	$1.1 \cdot 10^{-3}$	74.44
10^{-5}	10^{-2}	35541	174	742	$1.5 \cdot 10^{-4}$	110.58
10^{-5}	10^{-3}	35541	516	997	$1.2 \cdot 10^{-4}$	142.13
10^{-5}	10^{-4}	35541	801	1022	$1.1 \cdot 10^{-4}$	144.21
10^{-8}	10^{-2}	90742	174	742	$1.1 \cdot 10^{-4}$	278.92
10^{-8}	10^{-3}	90742	516	984	$1.2 \cdot 10^{-5}$	336.18
10^{-8}	10^{-4}	90742	801	1014	$7.7 \cdot 10^{-7}$	342.97
10^{-8}	10^{-5}	90742	973	1019	$9.9 \cdot 10^{-8}$	343.74
10^{-8}	10^{-6}	90742	1017	1024	$9.5 \cdot 10^{-8}$	343.80

FIG. 3.13 – *Comparaison de l'erreur avec la solution de référence et temps de calcul (10000 itérations) en fonction des seuils sur l'opérateur et la solution (équation de transport, condition initiale turbulente,méthode standard)*

ϵ_B	ϵ_U	# coef $B_2 > \epsilon_B$	# coef $U_0 > \epsilon_U \|U_0\|$	# coef $U_F > \epsilon_U \|U_F\|$	$\frac{\|U_F - Y_1\|_2}{\|U_F\|_2}$	temps (s)
10^{-2}	10^{-2}	1024	563	1080	$1.3 \cdot 10^{-3}$	103.72
10^{-2}	10^{-3}	1024	1280	1360	$1.3 \cdot 10^{-3}$	109.09
10^{-3}	10^{-2}	9193	563	1944	$1.3 \cdot 10^{-3}$	198.60
10^{-3}	10^{-3}	9193	1280	1985	$1.4 \cdot 10^{-3}$	201.04
10^{-4}	10^{-2}	17474	563	1872	$1.0 \cdot 10^{-3}$	252.48
10^{-4}	10^{-3}	17474	1280	2033	$9.9 \cdot 10^{-4}$	281.62
10^{-5}	10^{-2}	22709	563	1604	$2.0 \cdot 10^{-4}$	261.72
10^{-5}	10^{-3}	22709	1280	2008	$4.1 \cdot 10^{-5}$	336.10
10^{-5}	10^{-4}	22709	1669	2043	$3.5 \cdot 10^{-5}$	340.85
10^{-8}	10^{-2}	30917	563	1597	$1.9 \cdot 10^{-4}$	320.68
10^{-8}	10^{-3}	30917	1280	2009	$2.2 \cdot 10^{-5}$	414.00
10^{-8}	10^{-4}	30917	1669	2030	$2.0 \cdot 10^{-6}$	421.68
10^{-8}	10^{-5}	30917	1939	2041	$1.1 \cdot 10^{-7}$	422.97
10^{-8}	10^{-6}	30917	2028	2046	$2.1 \cdot 10^{-8}$	423.40

FIG. 3.14 – *Comparaison de l'erreur avec la solution de référence et temps de calcul (10000 itérations) en fonction des seuils sur l'opérateur et la solution (équation de transport, condition initiale turbulente, méthode BCR)*

3.2.5 Erreur et temps de calcul en fonction de l'ordre de l'ondelette

Dans cette partie les seuils ϵ_B et ϵ_U sont choisis égaux à 10^{-4} et 10^{-5}. Nous étudions la variation du nombre de degrés de liberté des systèmes (3.8) et (3.10) en fonction de l'ordre r de l'ondelette de base.

Pour la méthode BCR, l'ordre de l'ondelette influe peu sur le nombre de coefficients significatifs dans les systèmes (3.8) et (3.10) (et donc sur le temps de calcul), avec toutefois un meilleur compromis pour l'ordre 4. Les temps de calcul sont par ailleurs bien plus importants (entre deux et quinze fois plus important) que pour la méthode standard.

Nous nous contenterons donc d'exposer les résultats obtenus par la méthode standard. Le temps de calcul étant fortement lié à la représentation de la solution, celui-ci dépend de la régularité de l'ondelette choisie. Les tableaux résument le nombre de paramètres mis en jeu dans les opérateurs et la solution, l'erreur avec la solution de référence et le temps de calcul pour un ordre de l'ondelette variant de $r = 1$ à $r = 10$.

3.2.5.1 Équation de la chaleur – Méthode standard

Pour cette équation, d'après la figure 3.11, les seuils sur la matrice B_1 et sur le vecteur sont choisis de manière à avoir une erreur de l'ordre de 10^{-7}, soit $\epsilon_B = 10^{-4}$ et $\epsilon_U = 10^{-3}$.

ordre	# coef $B_1 > \epsilon_B$	# coef $U_0 > \epsilon_U \|U_0\|$	# coef $U_F > \epsilon_U \|U_F\|$	$\frac{\|U_F - Y_1\|_2}{\|U_F\|_2}$	temps (s)
1	42747	130	1022	$2.4 \cdot 10^{-6}$	1.70
2	16531	23	456	$3.3 \cdot 10^{-7}$	0.54
3	11219	33	198	$3.0 \cdot 10^{-7}$	0.24
4	9259	40	100	$1.7 \cdot 10^{-7}$	0.14
5	8703	46	66	$2.1 \cdot 10^{-7}$	0.10
6	7855	45	60	$1.7 \cdot 10^{-7}$	0.09
7	8179	51	62	$1.2 \cdot 10^{-7}$	0.09
8	8215	58	48	$1.6 \cdot 10^{-7}$	0.09
9	7183	64	32	$1.0 \cdot 10^{-7}$	0.07
10	7331	64	30	$1.9 \cdot 10^{-7}$	0.07

FIG. 3.15 – *Évolution du nombre de coefficients de la matrice et du vecteur supérieurs aux seuils, et du temps de calcul en fonction de l'ordre de l'ondelette (opérateur de diffusion)*

On constate que pour une équation régularisante comme la chaleur, il est intéressant d'utiliser des ondelettes d'ordre élevé, la solution étant infiniment dérivable : en effet les coefficients décroissent d'autant plus vite en échelle que le nombre de moments nuls est élevé. L'opérateur est aussi mieux représenté avec des ondelettes très régulières. On a également une diminution du nombre de degrés de liberté au cours du temps (voir figure 3.17).

3.2.5.2 Équation de transport – Méthode standard

Pour cette équation, les seuils précédents ont été conservés, soit : $\epsilon_B = 10^{-4}$ et $\epsilon_U = 10^{-3}$.

ordre	# coef $B_2 > \epsilon_B$	# coef $U_0 > \epsilon_U \|U_0\|$	# coef $U_F > \epsilon_U \|U_F\|$	$\frac{\|U_F - Y_1\|_2}{\|U_F\|_2}$	temps (s)
1	12244	777	1018	$1.0 \cdot 10^{-3}$	59.04
2	17975	630	1012	$1.1 \cdot 10^{-3}$	71.68
3	18509	538	1013	$1.1 \cdot 10^{-3}$	73.63
4	18996	516	1011	$1.1 \cdot 10^{-3}$	75.47
5	18343	502	1011	$1.1 \cdot 10^{-3}$	73.78
6	17848	475	1007	$9.9 \cdot 10^{-4}$	70.43
7	17422	466	1002	$9.6 \cdot 10^{-4}$	68.51
8	17545	472	1004	$9.2 \cdot 10^{-4}$	68.84
9	16861	438	1004	$8.8 \cdot 10^{-4}$	66.49
10	17495	439	1002	$9.4 \cdot 10^{-4}$	68.30

FIG. 3.16 – *Évolution du nombre de coefficients de la matrice et du vecteur supérieurs aux seuils, et du temps de calcul en fonction de l'ordre de l'ondelette (opérateur de transport)*

On constate que l'absence de régularité de la solution augmente considérablement le nombre de coefficients d'ondelettes significatifs et supprime la variation du nombre de ces coefficients en fonction de l'ordre de l'ondelette. Le nombre de degrés de liberté au cours du temps est oscillant, mais sa moyenne est stable (figure 3.17).

3.2.6 Conclusion

Dans cette section nous avons comparé deux méthodes de décompositions en ondelettes pour la résolution numérique de l'équation de la chaleur et de l'équation de transport unidimensionnelles, périodiques. Dans les deux cas, nous avons calculé le noyau de Green discret, puis décomposé celui-ci soit sur une base d'ondelettes standard, soit par la méthode BCR. Pour l'opérateur de diffusion périodique, nous avons optimisé le calcul de la décomposition (coût : $O(\log_2 N)$ opérations en standard, $O(N)$ opérations en non standard pour une matrice d'ordre N) ainsi que son stockage ($O(N \log_2^2 N)$ coefficients pour la forme standard, $O(N)$ coefficients pour la forme non standard). Dans le calcul de la solution, nous tenons compte de la lacunarité de l'opérateur et de celle de la solution.

Les tests numériques donnent pour les deux équations des temps de calculs plus faibles pour la méthode standard, malgré une meilleure représentation des opérateurs par la décomposition BCR. Ceci s'explique par l'impossibilité de prendre en compte une lacunarité éventuelle de la solution dans la méthode BCR. Pour le passage aux dimensions supérieures, nous utiliserons donc la méthode standard.

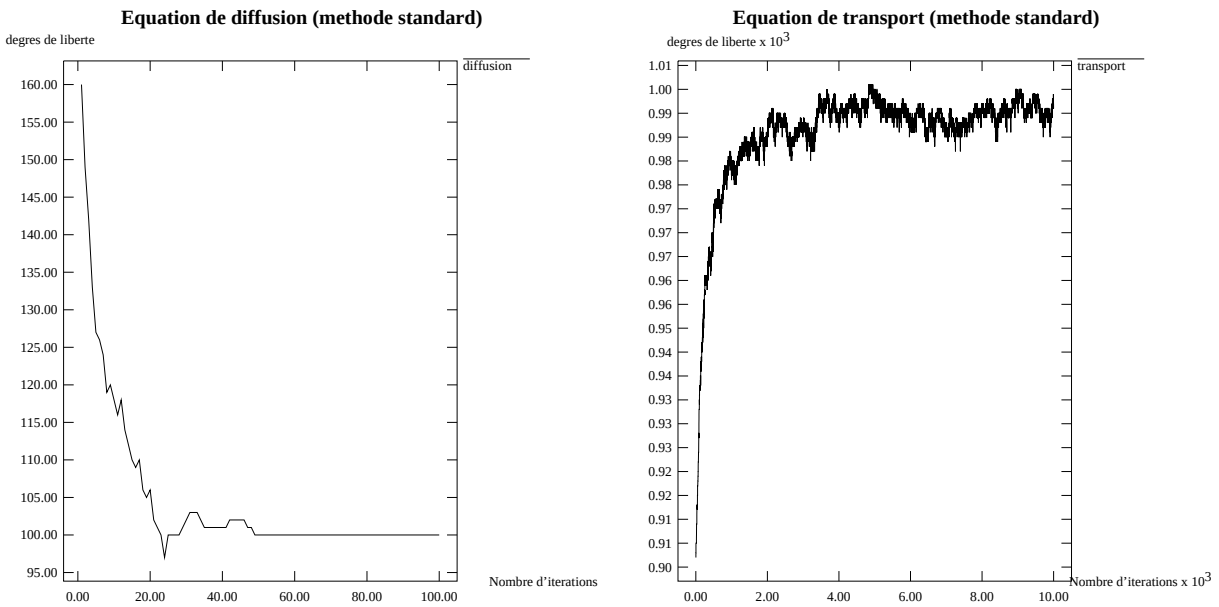


FIG. 3.17 – *Évolution du nombre de degrés de liberté (coefficients de la solution* $> 10^{-3}$*) en fonction du nombre d'itérations*

3.3 Expériences numériques en dimension deux

Dans cette partie nous nous proposons d'étendre la méthode développée au chapitre précédent aux dimensions supérieures. Nous présenterons des résultats en dimension deux, mais l'extension aux dimensions trois et plus est immédiate.

Nous nous proposons dans ce chapitre de résoudre l'équation de la chaleur bidimensionnelle :

$$\frac{\partial u}{\partial t} - \nu \Delta u = f(x, y), \tag{3.11}$$

sur le tore $\mathbb{T}^2$.

3.3.1 Factorisation de l'opérateur de la chaleur

Si nous appliquons directement la méthode précédente à l'équation (3.11), nous allons avoir à manipuler des matrices d'opérateurs bidimensionnels, soit des tableaux de taille N^4 (N représente le nombre de degrés de liberté par direction), ce qui est bien supérieur aux capacités de stockage des ordinateurs actuels si nous voulons garder la résolution $N = 1024$. Nous allons donc approcher cet opérateur par factorisation d'opérateurs **monodimensionnels** (inspirée de la méthode des directions alternées présentée dans [Peaceman et al.55]) :

$$\text{Pour } \alpha \text{ petit}, \qquad (1 - \alpha\Delta) \approx (1 - \alpha\frac{\partial^2}{\partial x^2})(1 - \alpha\frac{\partial^2}{\partial y^2}) \tag{3.12}$$

3.3.2 Schéma en temps

Soit δt un pas de discrétisation temporelle et $u^n(x,y) \approx u(x,y,n\delta t)$ une solution approchée de (3.11) au temps $n\delta t$. Deux schémas implicites en temps inspirés respectivement des schémas d'Euler et Crank-Nicolson sont appliqués à l'équation (3.11) qui devient en utilisant la factorisation (3.12) :

Schéma A :

$$(1-\nu\delta t\frac{\partial^2}{\partial x^2})(1-\nu\delta t\frac{\partial^2}{\partial y^2})u^{n+1} = u^n + \delta t f \tag{3.13}$$

Schéma B :

$$(1-\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1-\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2})u^{n+1} = (1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2})u^n + \delta t f \tag{3.14}$$

Théorème 8 *Les schémas A et B sont inconditionnellement stables, de précision respective $O(\nu\delta t)$ et $O(\nu\delta t)^2$.*

Démonstration : La stabilité est une conséquence directe de la stabilité des schémas unidimensionnels. La précision du schéma (3.13) est de $O(\nu\delta t)$ comme le schéma d'Euler, l'approximation (3.12) étant également d'ordre 1. Il reste à examiner le cas du schéma (3.14).

On sait que le schéma de Crank-Nicolson traditionnel :

$$(1-\nu\frac{\delta t}{2}\Delta)u^{n+1} = (1+\nu\frac{\delta t}{2}\Delta)u^n \tag{3.15}$$

est d'ordre deux en temps. Appliquons la factorisation (3.12) :

$$\left[(1-\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1-\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2}) - \left(\nu\frac{\delta t}{2}\right)^2\frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}\right]u^{n+1}$$

$$= \left[(1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2}) - \left(\nu\frac{\delta t}{2}\right)^2\frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}\right]u^n$$

$$\begin{aligned}
&(1-\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1-\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2})u^{n+1} \\
&= (1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2})u^n + \left(\nu\frac{\delta t}{2}\right)^2\frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}(u^{n+1}-u^n) \\
&\quad \text{Utilisons (3.15)} \\
&= (1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial x^2})(1+\nu\frac{\delta t}{2}\frac{\partial^2}{\partial y^2})u^n + \left(\nu\frac{\delta t}{2}\right)^3\frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}\Delta(u^{n+1}+u^n)
\end{aligned}$$

Ce qui montre bien le résultat annoncé. □

3.3.3 Schéma en espace

On effectue ensuite une discrétisation de type différences finies sur les opérateurs $(1 \pm \alpha \frac{\partial^2}{\partial x^2})$ et $(1 \pm \alpha \frac{\partial^2}{\partial y^2})$. Si nous supposons que l'on a $\delta x = \delta y$, on a deux fois le même opérateur, l'un s'appliquant sur les colonnes du champ l'autre sur les lignes, ce qui revient soit à multiplier à gauche par la matrice de l'opérateur discrétisé, soit à multiplier à droite par la transposée de cette même matrice. Cette situation se produit pour toute discrétisation sur une base construite par produit tensoriel de deux bases monodimensionnelles identiques. Les problèmes continus (3.13) et (3.14) sont donc ramenés à des problèmes discrets qui s'écrivent sous la forme matricielle suivante :
Schéma A :

$$A_{\nu\delta t}\ U^{n+1}\ {}^tA_{\nu\delta t} = U^n + \delta t F, \tag{3.16}$$

Schéma B :

$$A_{\frac{\nu\delta t}{2}}\ U^{n+1}\ {}^tA_{\frac{\nu\delta t}{2}} = A_{-\frac{\nu\delta t}{2}}\ U^n\ {}^tA_{-\frac{\nu\delta t}{2}} + \delta t F, \tag{3.17}$$

où les coordonnées $U^n_{i,j}$ de U^n sont des approximations de $U(x_i, y_j)$ avec $x_i = \frac{i}{\delta x}$ et $y_j = \frac{j}{\delta y}$ et la matrice A_α est une approximation aux différences finies de $(1 - \alpha \frac{\partial^2}{\partial x^2})$.

Le schéma (3.16) est équivalent à la résolution de N systèmes linéaires d'ordre N sur les lignes, puis N systèmes sur les colonnes de la matrice U^{n+1}. Pour le schéma (3.17), il faut de plus effectuer deux produits de matrices de dimension $N \times N$.

3.3.4 Accélération de la résolution par utilisation des bases d'ondelettes

Nous avons montré à la section précédente 3.2, que l'analyse en ondelettes permet d'accélérer le calcul des produits matrice-vecteur effectués lors de la résolution de l'équation de la chaleur unidimensionnelle. Dans les équations (3.16) et (3.17), nous avons ramené notre problème bidimensionnel à un ensemble de problèmes unidimensionnels auxquels nous pouvons également appliquer les mêmes techniques. En effet, on effectue une analyse en ondelettes suivant la direction x, ce qui permet de comprimer la matrice A_α et les colonnes de la matrice U^n. Puis on effectue une analyse en ondelettes suivant la direction y, ce qui permet de comprimer la matrice ${}^tA_\alpha$ et les lignes de la matrice U^n. Notons Q_p la matrice de changement de base associée à la transformée en ondelettes **unidimensionnelle** introduite à la section 1.1.3. Pour exprimer les équations (3.16) et (3.17) dans cette nouvelle base, il suffit de remplacer U^n par $Q_p U^n\ {}^tQ_p$, A_α par $Q_p A_\alpha\ {}^tQ_p$ et F par $Q_p F\ {}^tQ_p$. Pour alléger le texte, les mêmes notations A_α, $U^n \cdots$ sont conservées pour représenter désormais les matrices sur bases d'ondelettes $Q_p A_\alpha\ {}^tQ_p$, $Q_p U^n\ {}^tQ_p \cdots$

De manière identique à la section précédente 3.2, nous inversons les matrices intervenant dans le membre de gauche des équations (3.16) et (3.17), ainsi que le produit de matrices $A^{-1}_{\frac{\delta t}{2}} A_{-\frac{\delta t}{2}}$ intervenant dans le schéma (3.17), d'où l'écriture des schémas A et B en bases d'ondelettes :

$$U^{n+1} = B\ U^n\ \ {}^tB + \delta t\ C\ F\ \ {}^tC \tag{3.18}$$

où $B = C = A^{-1}_{\delta t}$ pour le schéma A, $B = A^{-1}_{\frac{\delta t}{2}} A_{-\frac{\delta t}{2}}$, $C = A^{-1}_{\frac{\delta t}{2}}$ pour le schéma B. Une

itération se résume donc au calcul de deux produits et d'une somme de matrices pour le schéma A comme pour le schéma B.

Complexité théorique : la complexité théorique d'une itération est de $O(c\ P)$ opérations où c est le nombre moyen d'éléments considérés comme non nuls d'une ligne de la matrice B, et P le nombre de coefficients considérés comme non nuls de U^n. Ce chiffre est à comparer avec celui obtenu dans le cas des différences finies, à savoir $O(l\ N^2)$ opérations (par exemple pour une factorisation LU), où l est la largeur de bande de $A_{\delta t}$ et N l'ordre de la matrice U^n (voir section 3.3.6).

3.3.5 Accélération du schéma en temps

3.3.5.1 Calcul des puissances itérées de l'opérateur $(1 - \alpha\Delta)$

Les bases d'ondelettes possèdent une autre propriété spécifique : dans ces bases, la matrice itérée B^n de la matrice B intervenant dans le schéma (3.18), est beaucoup plus creuse que B, alors que le phénomène inverse se produit si l'on reste sur la base canonique (figure 3.18). En effet, sur cette figure, on constate qu'en différences finies, pour le schéma A comme pour le schéma B, le nombre de coefficients dans la matrices B^n augmente avec n ; décomposée en ondelettes, la matrice B^n a un nombre de coefficients significatifs qui diminue avec n dans le schéma A, ainsi que dans le schéma B, malgré une légère augmentation pour les n petits.

Ceci s'explique simplement en regardant les différentes valeurs propres de la matrice B qui sont $\frac{1}{1+\nu\delta t k^2}$ dans le cas du schéma A et $\frac{1-\frac{\nu\delta t}{2}k^2}{1+\frac{\nu\delta t}{2}k^2}$ dans le cas du schéma B ; le spectre de cette matrice est donc composé de 1 et de valeurs propres dont le module est inférieur à 1. Le vecteur propre associé à la valeur propre 1 est un vecteur ayant toutes ses composantes égales, ce vecteur faisant partie de la base d'ondelettes, le reste de la matrice tend vers 0 quand n tend vers l'infini.

3.3.5.2 Schéma accéléré

Ceci va permettre d'effectuer une accélération en temps du schéma (3.18) en ne calculant que les termes de la forme U^{2^p}. On a donc une première étape identique :

$$\begin{aligned} F^1 &= \delta t\ C\ F\ {}^tC \\ U^1 &= B\ U^0\ {}^tB + F^1 \end{aligned}$$

On ne calcule ensuite que les U^n de la forme U^{2^p} :

$$\text{Pour } p \geq 1, \left\{ \begin{array}{l} U^{2^{p+1}} = B^{2^p}\ U^{2^p}\ {}^tB^{2^p} + F^{2^p} \\ F^{2^{p+1}} = B^{2^p}\ F^{2^p}\ {}^tB^{2^p} + F^{2^p} \\ B^{2^{p+1}} = B^{2^p}\ B^{2^p} \end{array} \right. \tag{3.19}$$

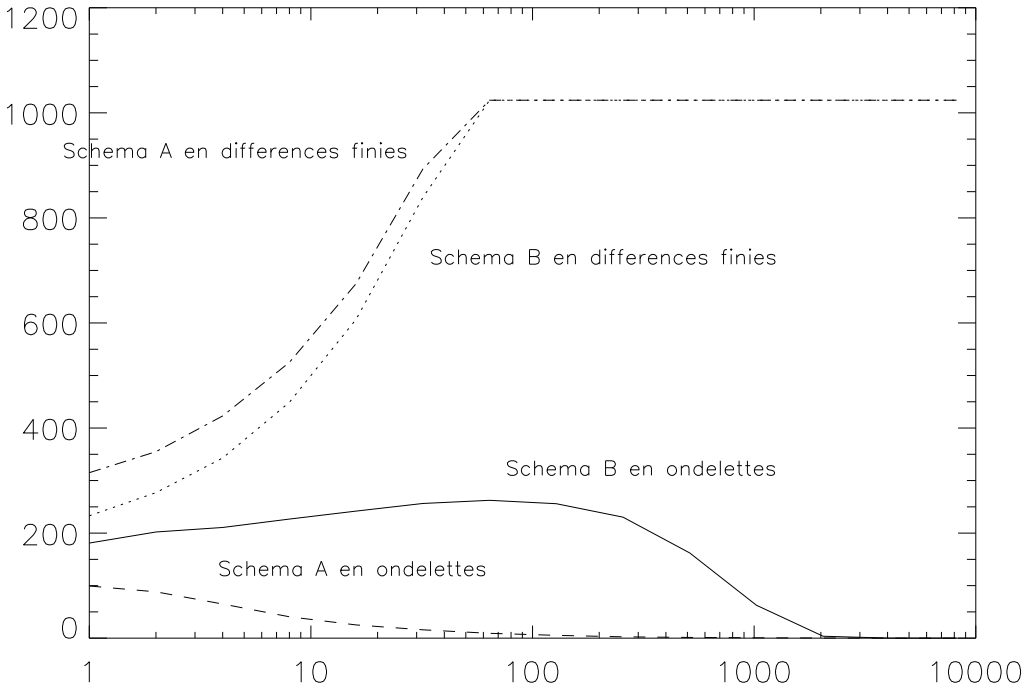


FIG. 3.18 – *Nombre moyen de coefficients par lignes dans la matrice B^n pour les schémas A et B, en ondelettes et en différences finies en fonction de n ($\nu\delta t = 10^{-5}, \epsilon = 10^{-8}$).*

Remarque : si $U^0 = 0, F^{2^p} = U^{2^p}$, on a donc à calculer à chaque étape uniquement $U^{2^{p+1}}$ et $B^{2^{p+1}}$. Pour calculer U^n, pour $n = 2^p$, nous n'avons donc à effectuer que p itérations, le temps de calcul de U^n se réduit alors à $O(\log(n)\ c\ P)$ opérations.

3.3.5.3 Application à la résolution de l'équation de Poisson

Cette nouvelle écriture du schéma (3.18) permet la résolution de l'équation de Poisson :

$$-\Delta u = f \tag{3.20}$$

par une méthode itérative en calculant la solution numériquement stationnaire de (3.19). Dans la pratique, le nombre d'itérations est lié au paramètre $\nu\delta t$: la solution de l'équation de la chaleur étant de la forme $(1 - e^{-8\pi^2\nu t})f(x, y)$, on peut considérer que l'on a atteint la solution stationnaire quand $t = \frac{1}{\nu}$. Une solution approchée de (3.20) est donc donnée par $U^{2^{Niter}}$, où $Niter = -\log_2(\nu\delta t)$ est le nombre d'itérations nécessaires pour obtenir la solution correspondant au temps $t = \frac{1}{\nu}$.

3.3.6 Résultats numériques

3.3.6.1 Équation de la chaleur bidimensionnelle

Les schémas (3.18) et (3.19) sont appliqués à l'équation :

$$\begin{cases} \frac{\partial u}{\partial t} - \nu \Delta u = \sin(2\pi x)\cos(2\pi y), \\ u(x,y,0) = 0 \\ \text{avec conditions aux limites périodiques en } x \text{ et } y \end{cases} \tag{3.21}$$

dont la solution exacte est $u(x,y,t) = \frac{1}{8\pi^2\nu}\left(1 - e^{-8\pi^2\nu t}\right) sin(2\pi x) cos(2\pi y)$.

Les tests numériques ont été effectués avec des ondelettes à support compact d'ordre quatre en espace construites dans [Daubechies92], et un nombre de points $N = 1024$ dans chaque direction. Dans les schémas (3.18) et (3.19), les coefficients de la matrice B inférieurs au seuil $\epsilon_B = 10^{-8}$, ainsi que ceux de U^n inférieurs au seuil $\epsilon_U = 10^{-5}\|U\|$ sont mis à 0.

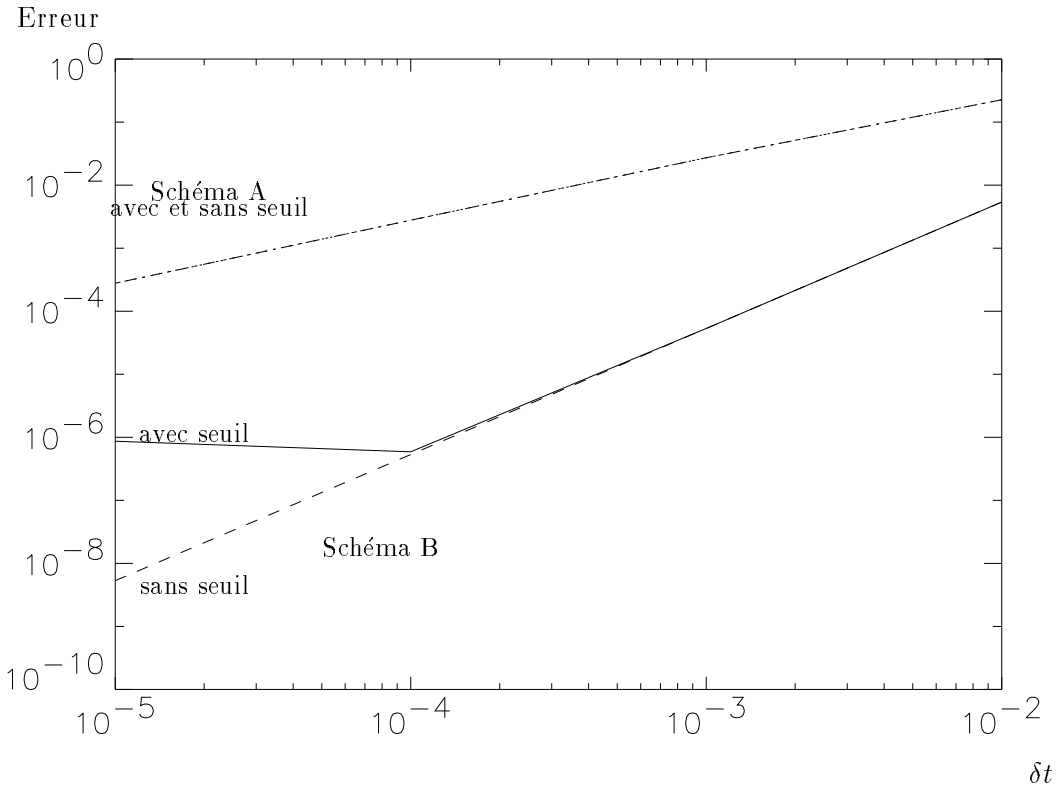


FIG. 3.19 – *Erreur l^2 relative entre la solution exacte de (1.10) et la solution calculée par les schémas A ou B, en fonction du pas de temps δt (avec et sans seuil sur les matrices).*

La figure 3.19 représente l'erreur à $t = 0.02$, $\nu = 1$, entre la solution exacte et les solutions approchées du schéma (3.18) suivant le pas de temps choisi, d'une part avec les seuils précédents sur les matrices, d'autre part sans seuil. Sur cette figure on constate que

pour le schéma A, d'ordre un en temps, la pente est exactement de un. L'approximation faite lors de l'utilisation du schéma aux directions alternées est donc d'ordre supérieur à l'approximation en temps, et ceci malgré la présence des seuils sur les matrices. Pour le schéma B, si on ne met pas de seuil sur les matrices, la pente est de deux, ce qui correspond bien à l'ordre du schéma en temps, ceci justifie une fois encore l'utilisation de la méthode des directions alternées. Si on utilise des seuils sur les matrices, la précision du schéma est limitée par ces seuils pour les pas de temps inférieurs à $\delta t = 10^{-4}$.

En fait, cette méthode permet de choisir la précision souhaitée, pour cela il faut choisir un pas de temps adapté (préférer le schéma B, d'ordre deux en temps, donc permettant des pas de temps beaucoup plus importants pour une même précision) et des seuils sur les matrices de l'ordre de l'erreur souhaitée . On peut donc avoir soit une solution peu précise en très peu de temps, soit choisir des paramètres permettant une précision bien meilleure au détriment du temps de calcul.

Complexité d'une itération : pour le schéma B, en prenant $\delta t = 10^{-5}$ et les seuils précédents, la complexité d'une itération est de $2cP$ opérations avec $c = 128$ et $P \approx 2500$ pour le calcul de U^1, à comparer avec 8×1024^2 pour la même méthode en différences finies. Cette complexité est réduite pour le calcul des U^{2^p} suivants : la lacunarité de l'opérateur itéré B^{2^p} du schéma (3.19) décroissant fortement avec p (figure 3.18), le nombre moyen c de coefficients par lignes de la matrice B^{2^p} est inférieur à 128 pour $p \geq 7$. On a même $c \ll 50$ pour $p > 10$ et $c < 1$ pour $p \geq 11$.

3.3.6.2 Équation de Poisson bidimensionnelle

La solution de l'équation de Poisson

$$\begin{cases} -\Delta u = f \\ \iint u \; dxdy = 0 \\ \text{avec conditions aux limites périodiques,} \end{cases} \tag{3.22}$$

est calculée par le schéma (3.19) d'ordre 2 : celui-ci fournit, quand $p \to +\infty$ et $\nu = 1$, une approximation de la solution stationnaire de (3.11).

Nous utilisons le même second membre qu'à la section précédente, à savoir : $f = \sin(2\pi x)\cos(2\pi y)$. La solution exacte est alors $u_{ex}(x,y) = \frac{1}{8\pi^2\nu} sin(2\pi x) cos(2\pi y)$.

Nous présentons une comparaison des temps de calculs (sur IBM RS 6000) pour différents pas de temps δt (seuil $\epsilon_B = 10^{-8}$ sur la matrice B , seuil $\epsilon_U = 10^{-5}$ sur la solution U^n, permettant d'atteindre une erreur l^2 de l'ordre de 10^{-7}, dimension d'espace $N^2 = 1024^2$) :

δt	Niter.	Stockage	temps d'initialisation (s)	temps de calcul (s)	Erreur l^2
$1\ 10^{-4}$	13	2459263	38.82	4.22	$1.15\ 10^{-6}$
$1\ 10^{-5}$	16	1124561	19.58	4.28	$2.29\ 10^{-7}$
$5\ 10^{-6}$	17	1050549	17.95	4.18	$1.35\ 10^{-7}$
$1\ 10^{-6}$	19	1198221	20.32	4.84	$1.38\ 10^{-7}$
$1\ 10^{-7}$	23	1530655	24.54	6.72	$6.02\ 10^{-7}$

– δt : pas de temps utilisé dans le schéma (3.19).

- Niter. est le nombre d'itérations pour atteindre la solution stationnaire « numérique ».
- stockage : nombre total de coefficients effectivement utilisé des matrices $(B^p)_{1\leq p\leq \text{Niter}}$ (le stockage de la solution est comme précédemment de l'ordre de 2500 coefficients).
- temps d'initialisation : temps nécessaire au calcul des matrices $(B^p)_{1\leq p\leq \text{Niter}}$.
- temps de calcul : temps nécessaire au calcul des $(U^p)_{1\leq p\leq \text{Niter}}$.
- Erreur l^2 : Erreur l^2 relative entre la solution calculée et la solution exacte U_{ex}.

Remarque : Pour déterminer Niter., il suffit de contrôler le nombre de coefficients significatifs restant dans la matrice B, la solution stationnaire est atteinte quand ce nombre est nul.

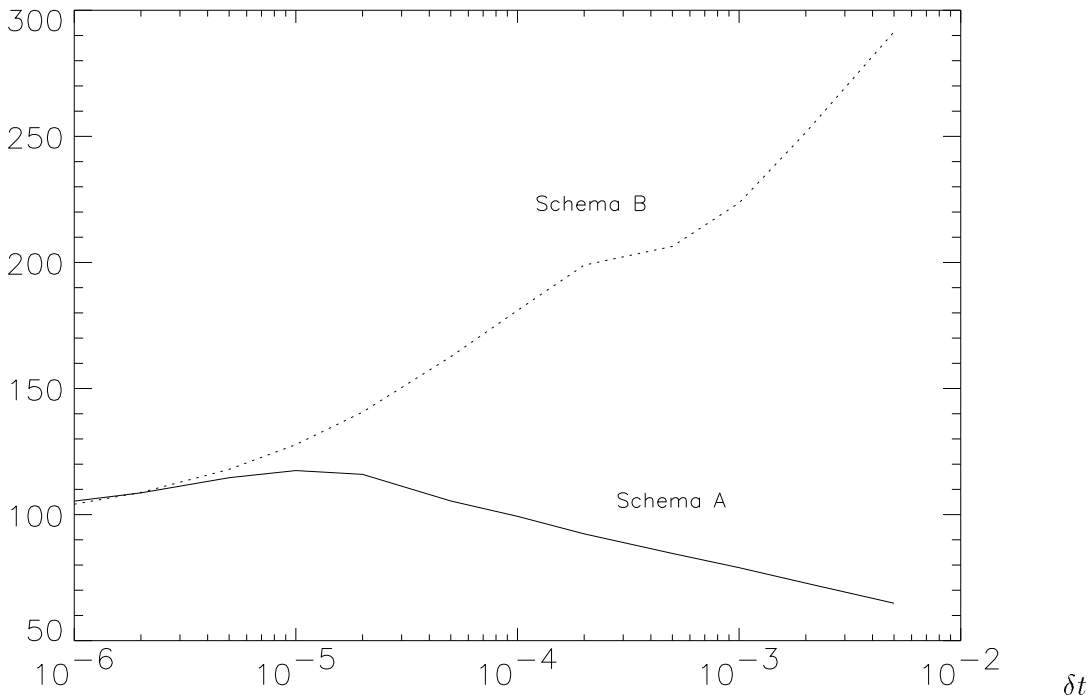


FIG. 3.20 – *Nombre moyen de coefficients par ligne dans la matrice B pour les schémas A et B en fonction du pas de temps δt*

Il faut noter que plus δt est petit, plus le nombre d'itérations augmente (Niter.$= -\log_2(\delta t)$), mais au début le temps de calcul diminue, la matrice B étant plus creuse (voir figures 3.20 et 3.21). En effet, plus le pas de temps est petit, plus l'opérateur se rapproche de l'identité, le nombre de coefficients nécessaires pour représenter sa discrétisation est

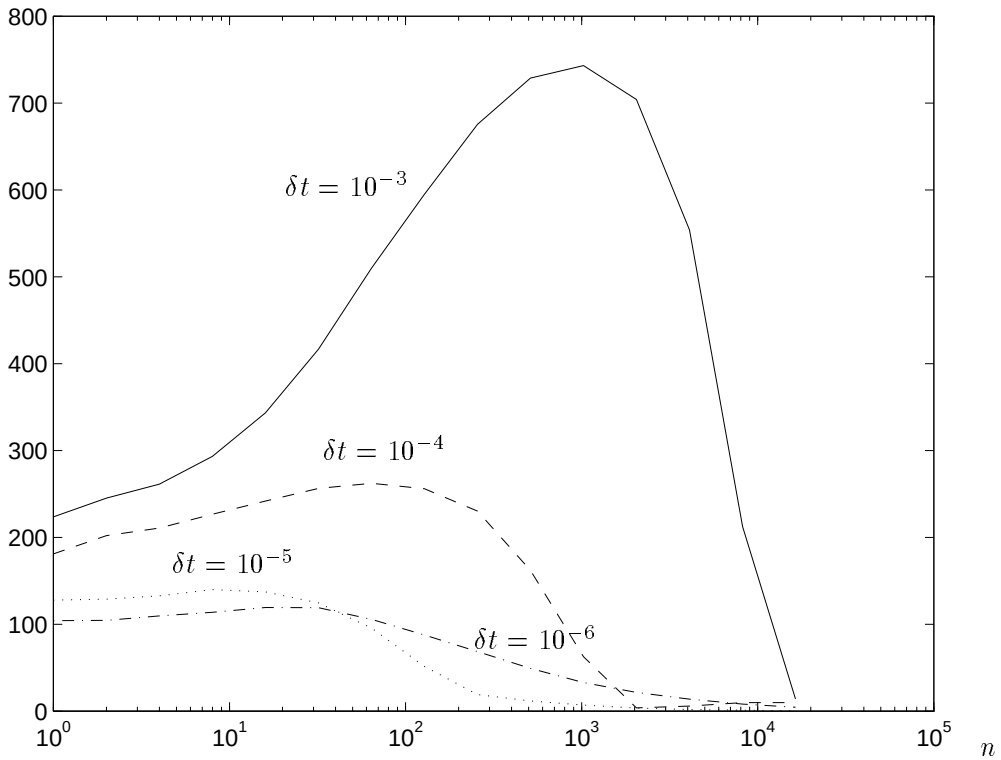


FIG. 3.21 – *Nombre moyen de coefficients par ligne dans la matrice B^n en fonction de n avec différents pas de temps pour le schéma B*

donc plus faible. Pour déterminer le pas de temps le plus efficace, il suffit de regarder le stockage nécessité par les matrices B^p, d'après la complexité théorique calculée à la section précédente, le temps de calcul est proportionnel à ce stockage, ceci est vérifié par le tableau ci-dessus. L'erreur l^2 dépend essentiellement des seuils ϵ_B et ϵ_U et varie donc peu avec δt (nous utilisons ici des pas de temps suffisamment petits pour que le facteur limitant la précision soit les seuils sur les matrices).

Nous allons à présent utiliser cette méthode, pour calculer la fonction de courant ψ associée à un champ de vorticité ω *f10a1n40* déjà présenté à la section 1.2.2. Pour cela, il faut résoudre l'équation : $\Delta\psi = \omega$ sur $[0,1]^2$ avec conditions aux limites périodiques. La fonction de courant correspondant à ce champ de vorticité est représentée figure 3.22. Le nombre de coefficients significatifs de la solution est de moins de 10 % du nombre total de coefficients. Ce calcul a nécessité 17 itérations effectuées en 84 s sur IBM RS 6000, l'écart avec une solution numérique calculée par la méthode de Fourier est de 10^{-8}, les paramètres δt, ϵ_B et ϵ_U utilisés sont ceux de la troisième ligne du tableau.

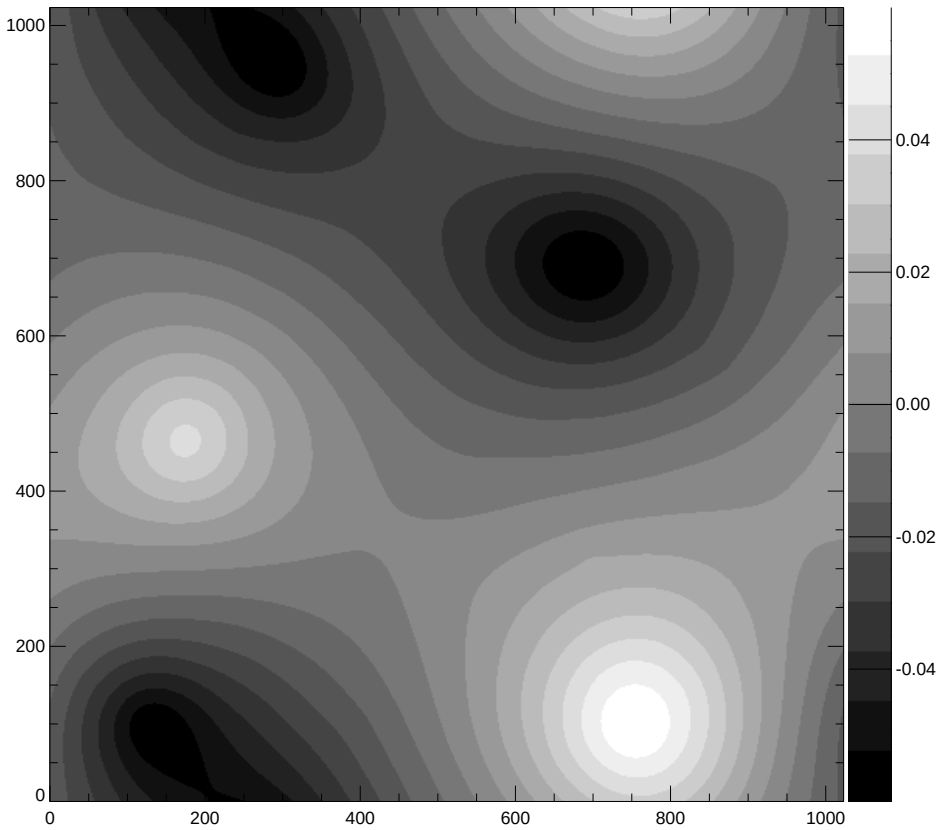


FIG. 3.22 – *Fonction de Courant associée au champ* f10a1n40

3.4 Équation de Navier-Stokes bidimensionnelle

À partir des résultats de la section précédente nous allons élaborer un schéma numérique « pseudo-ondelette » (par analogie avec les méthodes pseudo-spectrales) pour résoudre les équations de Navier-Stokes incompressibles en dimension deux.

3.4.1 Présentation des équations

La formulation de ces équations dans le cas général est la suivante :

$$\begin{cases} \dfrac{\partial \overrightarrow{v}}{\partial t} + (\overrightarrow{v} \cdot \nabla)\overrightarrow{v} = \overrightarrow{F} - \dfrac{1}{\rho}\nabla p + \nu \Delta \overrightarrow{v} \\ \\ \nabla \cdot \overrightarrow{v} = 0 \\ + \text{ des conditions initiales et des conditions aux limites} \end{cases} \tag{3.23}$$

où $\overrightarrow{v}$ est la vitesse, ρ la masse volumique du fluide, $\overrightarrow{F}$ les forces extérieures agissant sur le fluide, p la pression et ν la viscosité.

Rappelons que les conditions que doit remplir un fluide pour être considéré comme incompressible sont les suivantes ([Landau et al.59], page 24) :

$$\begin{cases} v \ll c \\ \tau \gg l/c, \end{cases}$$

où c est la vitesse du son dans le fluide, v un ordre de grandeur de la vitesse du fluide, et où τ et l sont de l'ordre de grandeur des temps et distances pour lesquels la vitesse v subit une modification appréciable. La dernière inégalité signifie que les interactions entre les différentes parties du fluides peuvent être considérées comme instantanées, et donc la vitesse du son infinie.

En prenant le rotationnel de l'équation (3.23), nous obtenons la formulation à l'aide des variables vorticité / fonction de courant s'écrivant sous la forme :

$$-\Delta\psi = \omega \tag{3.24}$$

$$\frac{\partial \omega}{\partial t} + J(\psi, \omega) = \nu\Delta\omega + \mathrm{rot} f \tag{3.25}$$

La vorticité étant définie à partir de la vitesse par l'équation :

$$\omega = \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y}. \tag{3.26}$$

La fonction de courant ψ est reliée à la vitesse par la relation :

$$\overrightarrow{v} = (-\frac{\partial \psi}{\partial y}, \frac{\partial \psi}{\partial x}), \tag{3.27}$$

et J représente le jacobien :

$$J(\psi,\omega) = \frac{\partial\psi}{\partial x}\frac{\partial\omega}{\partial y} - \frac{\partial\psi}{\partial y}\frac{\partial\omega}{\partial x} \ . \tag{3.28}$$

Lorsque la viscosité est nulle, deux quantité sont invariante : L'énergie ($E = \frac{1}{2}\iint \overrightarrow{v}^2 dx\ dy$) et l'enstrophie ($Z = \frac{1}{2}\iint \omega^2 dx\ dy$). Lors de l'analyse des résultats, on utilisera les spectres de Fourier de ces deux grandeurs définis par ($\widehat{\overrightarrow{v}}$ représente la transformée de Fourier de $\overrightarrow{v}$, $\widehat{\omega}$, celle de ω) :

$$E(k) = \frac{1}{2}\int_0^{2\pi} \widehat{\overrightarrow{v}}^2(k,\theta)\ d\theta \tag{3.29}$$

$$Z(k) = \frac{1}{2}\int_0^{2\pi} \widehat{\omega}^2(k,\theta)\ d\theta \tag{3.30}$$

3.4.2 Schéma en ondelettes pour Navier-Stokes

Le schéma de résolution que nous proposons pour résoudre ces équations dans le cas bidimensionnel bipériodique incompressible est basé sur un schéma explicite en temps pour le terme d'advection –Le schéma d'Adams-Bashforth–, implicite pour le terme de diffusion –le schéma de Crank-Nicolson– ; les termes non linéaires étant calculés par une méthode de collocation. La discrétisation du Jacobien utilisée est celle proposée par Arakawa dans [Arakawa88]. Elle consiste à écrire le jacobien sous la forme :

$$J(\psi,\omega) = \frac{1}{3}\left((\frac{\partial\psi}{\partial x}\frac{\partial\omega}{\partial y} - \frac{\partial\omega}{\partial y}\frac{\partial\psi}{\partial x}) - (\frac{\partial}{\partial x}(\omega\frac{\partial\psi}{\partial y}) - \frac{\partial}{\partial y}(\omega\frac{\partial\psi}{\partial x})) + (\frac{\partial}{\partial x}(\psi\frac{\partial\omega}{\partial y}) - \frac{\partial}{\partial y}(\psi\frac{\partial\omega}{\partial x}))\right)$$

Les dérivées partielles nécessaires au calcul de ce jacobien sont ensuite discrétisées par un schéma centré aux différences finies d'ordre 2. Écrit sous cette forme, lorsque la diffusion est nulle, le schéma conserve à la fois l'entrophie et l'énergie.

Nous notons :

- $\tilde{\omega}$ quand la grandeur est représentée en base d'ondelettes (nous continuons à utiliser les ondelettes à support compact de Daubechies d'ordre quatre),
- ω quand la grandeur est représentée en base nodale,
- N, le nombre de points dans chaque direction pour la grille nodale associée à l'échelle la plus fine $N = 2^p$,
- N_{Btot}, le nombre total d'éléments non nuls dans les matrices $N \times N$ utilisées pour la résolution de l'équation de Poisson. Ces matrices sont les puissances itérées de B, voir équation (3.19). N_{Btot} est de l'ordre de $0.5N^2$ pour $N = 1024$, mais sa valeur peut être supérieure à N^2 ; son comportement asymptotique est en $O(N\log(N))$. Dans le cas des ondelettes à support compact, la constante est proportionnelle à l'ordre des ondelettes utilisées, voir section 3.2.3.

- N_{B1}, le nombre d'éléments non nuls dans la matrice $N \times N$ utilisée pour la résolution de l'équation de la chaleur (de l'ordre de $0.05N^2$ pour $N = 1024$, son comportement asymptotique est également en $O(N \log(N))$, la constante étant aussi proportionnelle à l'ordre de l'ondelette),

- N_X, le nombre de coefficients d'ondelettes utilisés pour représenter la solution discrétisée sur $N \times N$ points. Ce nombre est très variable en fonction de la structure du champ. Par exemple, sa valeur est comprise entre $0.05N^2$ et $0.5N^2$ pour des champs turbulents 1024×1024. Son comportement asymptotique dépend de la régularité du champ. Si l'on suppose le champ Hölderien partout sauf en un nombre fini de points, le comportement asymptotique est en $O(\log(N))$. En effet, chaque singularité crée à chaque échelle un nombre constant de coefficients non nuls et le nombre d'échelles est de $\log(N)$; le reste des coefficients est inférieur au seuil choisi pour une échelle suffisamment petite d'après le théorème 1, voir section 1.2.

Le schéma s'écrit alors de la manière suivante :

Partant de $\tilde{\omega}^n$,

1. on calcule $\tilde{\psi}^n$ par la formule : $\Delta\tilde{\psi}^n = \tilde{\omega}^n$ à l'aide du schéma (3.19), la complexité de cette étape est de $O(N_{Btot} \times N_X/N)$ opérations.

2. On effectue ensuite deux transformées en ondelettes inverses :

$$\psi^n = FWT^{-1}(\tilde{\psi}^n),\ \omega^n = FWT^{-1}(\tilde{\omega}^n),$$

   la complexité est donc de $O(N^2)$ opérations.

3. On calcule $S^n = \delta t\ J(\psi^n, \omega^n) + F^n$ (où F^n est le forçage éventuel) par un schéma aux différences finies, le coût de cette étape est aussi en $O(N^2)$ opérations.

4. On effectue une transformée en ondelettes directe :

$$\tilde{S}^n = FWT(S^n),$$

   le coût de cette étape est également en $O(N^2)$ opérations.

5. Enfin, on résout l'équation de la chaleur en utilisant le schéma (3.17) :

$$A_{\nu\frac{\delta t}{2}}\tilde{\omega}^{n+1} = A_{-\nu\frac{\delta t}{2}}\tilde{\omega}^n + \frac{3}{2}\,\tilde{S}^n - \frac{1}{2}\,\tilde{S}^{n-1}$$

   Cette étape a une complexité de $O(N_{B1} \times N_X/N + N^2)$ opérations.

Si l'on effectue la somme des complexités de chaque étape, on obtient le résultat :

$$O(\frac{N_{Btot}N_X}{N}) + O(N^2) + O(N^2) + O(N^2) + O(\frac{N_{B1}N_X}{N} + N^2)$$

Si l'on utilise le comportement asymptotique indiqué plus haut pour N_{Btot}, N_{B1} et N_X, cette somme devient :

$$O(\frac{N\log(N)\log(N)}{N}) + O(N^2) + O(N^2) + O(N^2) + O(\frac{N\log(N)\log(N)}{N} + N^2).$$

Les termes prépondérants sont ceux en $O(N^2)$; la complexité théorique totale de cette méthode est par conséquent en $O(N^2)$ opérations. Pour calculer la constante devant ce terme de plus haut degré, examinons chaque terme de cet ordre :

- La transformée en ondelettes nécessite 2 LN^2 opérations, L étant la longueur des filtres utilisés (ici $L = 8$) ; le schéma en utilise trois à chaque étape.
- Le calcul du Jacobien par la méthode d'Arakawa nécessite 9 ON^2 opérations O étant l'ordre du schéma aux différences finies utilisé pour le calcul des dérivées (ici $O = 2$).
- Le terme en $O(N^2)$ intervenant dans l'étape 5 est issu du calcul de la somme du terme $\tilde{S}^n$ et du terme $\tilde{S}^{n-1}$ qui nécessite donc 2 N^2 opérations.

Le coût total de la méthode est donc de $(6\ L + 9\ O + 2)N^2$ opérations par itération (dans la simulation présentée 68 N^2 opérations). Cette méthode est donc d'une complexité légèrement inférieure à celle d'une méthode spectrale (en $O(N^2\log(N))$. La différence de complexité est dûe à la complexité inférieure de la transformée en ondelettes rapide (en $O(N^2)$ en dimension deux, alors que la FFT est en $O(N^2\log(N))$ en dimension deux). Mais, la constante étant plus importante dans le cas de la méthode en ondelettes, la différence ne se fait sentir que pour les très grandes valeurs de N. Lors de l'utilisation d'un schéma implicite en temps, les méthodes type éléments finis sont elles aussi, au minimum d'une complexité de $O(N^2\log(N))$ opérations (par exemple lors de l'utilisation de la méthode des directions alternées, le calcul de complexité se trouve dans [Peaceman et al.55]).

3.4.3 Résultats numériques

3.4.3.1 Simulation basse résolution

Nous allons tout d'abord donner les résultats d'une simulation basse résolution effectuée sur station de travail afin de valider le modèle utilisé. Nous prenons comme condition initiale le champ *initialfred* que nous avons déjà présenté à la section 1.2.2. Nous donnons à la figure 3.23 une représentation dans l'espace physique du champ de vorticité ainsi que le spectre en Fourier de l'enstrophie.

Les paramètres de la simulation sont les suivants :

- la viscosité ν est prise égale à 10^{-3}
- le pas de temps δt est de 10^{-4}.
- On utilise des ondelettes d'ordre 4 et un schéma aux différences finies d'ordre 4 pour la discrétisation en espace ; le nombre de points de discrétisation est de $N = 128$ dans chaque direction.

– Les seuils sont $\epsilon_B = 10^{-7}$ pour la matrice de l'opérateur et $\epsilon_U = 10^{-4}$ pour le champ, c'est à dire les seuils précédemment utilisés diminués d'un ordre de grandeur, la résolution ayant été diminuée d'un facteur huit.

Les solutions aux temps $t = 1.$ et $t = 2.$ sont données aux figures 3.24 et 3.25. Sur ces figures on constate que l'on a bien le comportement décrit par exemple dans [Santangelo et al.89], c'est à dire que l'on a une organisation du champ en structures cohérentes de grandes échelles. Le champ au temps $t = 2.$ est dominé par un tourbillon dont l'échelle est limitée par la taille de la « boîte » utilisée pour la simulation.

Dans cette simulation à faible résolution, la compression obtenue sur le champ est relativement faible : on utilise entre 65 % et 99 % des coefficients d'ondelettes disponibles (99 % pour la condition initiale 65 % pour la solution finale, la diminution étant linéaire en fonction du temps). Le stockage de la matrice $N \times N$ utilisée dans la résolution de l'équation de la chaleur nécessite $N_{B1} = 5868$ coefficients (à comparer avec $N^2 = 16384$) ; le stockage des matrices $N \times N$ utilisées pour résoudre l'équation de Poisson nécessitent $N_{Btot} = 141422$, l'équation étant résolue en 22 itérations (on utilise donc 22 matrices pour résoudre l'équation de Poisson, la résolution étant relativement faible, la compression obtenue sur ces 22 matrices n'est pas très élevée, le nombre total de coefficients est donc supérieur à N^2).

Le temps de calcul d'une itération varie au cours du temps : en effet, le taux de compression sur la solution n'est pas le même, il est donc plus rapide de calculer les dernières itérations. Pour les premières itérations, le temps de calcul par itération est de 10 secondes, alors que pour les dernières itérations il est de seulement 6 secondes (cette simulation a été effectuée sur DEC ALPHA).

3.4.3.2 Comparaison avec un schéma spectral sur une simulation haute résolution

Nous présentons ici une simulation numérique haute résolution de l'interaction de trois tourbillons.

Description du champ La résolution est de 1024^2 et l'écoulement est non forcé. La taille de la boite est de $2\pi \times 2\pi$. L'état initial (figure 3.26) est composé de trois tourbillons, deux tourbillons tournent dans le sens direct, le troisième dans le sens inverse. Le profil des tourbillons suit une loi gaussienne : $\omega = Ae^{-r^2/\sigma^2}$. Les paramètres des trois tourbillons sont les suivants :

tourbillon 1 position : $(\frac{3\pi}{4}, \pi)$, amplitude : $A = \pi$, écart type : $\sigma = \frac{1}{\pi}$,

tourbillon 2 position : $(\frac{5\pi}{4}, \pi)$, amplitude : $A = \pi$, écart type : $\sigma = \frac{1}{\pi}$,

tourbillon 3 position : $(\frac{5\pi}{4}, \pi + \frac{\pi}{2\sqrt{2}})$, amplitude : $A = -\frac{\pi}{2}$, écart type : $\sigma = \frac{1}{\pi}$,

Le pas de temps est de $\delta t = 2.5\ 10^{-3}$ et la viscosité de $\nu = 5\ 10^{-5}$.

Le but de cette simulation est d'étudier l'appariement de deux tourbillons de même signe. Le troisième tourbillon de signe opposé est placé de manière à forcer la collision des deux premiers. Cette simulation a été réalisée par N. Kevlahan à l'aide d'un schéma pseudo-spectral. Celui-ci utilise la décomposition de Fourier partout où celle-ci est utilisable : pour l'inversion du Laplacien nécessaire au calcul de la fonction de courant, l'inversion du Laplacien nécessaire au calcul du terme de diffusion de manière implicite et le calcul des dérivées partielles du jacobien. Le produit des dérivées partielles est ensuite calculé dans l'espace physique. Afin d'éviter un éventuel recouvrement du spectre provoqué par ce terme quadratique, les modes de Fourier compris dans l'intervalle $[\frac{2}{3}k_{max}, k_{max}]$ sont mis à zéro. L'évolution du champ issu de cette simulation est représentée à la figure 3.28.

Dès le temps $t = 10.$, on constate le début de l'appariement des deux tourbillons, qui ont totalement fusionnés au temps $t = 40$, le troisième tourbillon restant relativement indépendant de cette interaction. Sur les spectres d'enstrophie et d'énergie (figure 3.30 et 3.32), on remarque la création de petites échelles lors de la fusion qui disparaissent progressivement sous l'effet de la dissipation dûe à la viscosité.

Comparaison de la simulation en ondelettes avec la simulation de référence Lors de l'utilisation de la méthode en ondelettes, nous avons utilisé un seuil de $\epsilon_B = 10^{-10}$ sur la matrice de l'opérateur et un seuil de $\epsilon_U = 10^{-8}$ sur le champ.

Dans l'espace physique, au temps $t = 10.$, on constate la création de petites irrégularités aux extrémités des tourbillons (voir figure 3.29). Le champ devenant plus complexe, ces irrégularités ne sont plus visibles ensuite. Celles-ci ne créent pas de différences importantes au niveau des spectres (voir figures 3.31 et 3.33) : les petites échelles représentent une énergie très voisine dans les deux simulations. En effet, les spectres d'enstrophie et d'énergie des deux méthodes sont extrêmement proches. Ceci permet de montrer la bonne précision de la méthode utilisée.

La simulation utilisant le code en ondelettes a nécessité 1152 heures de calcul sur IBM RS 6000. La simulation à l'aide du code spectral, réalisée sur Cray C90 est effectuée en 6186 secondes (le rapport de puissance de calcul entre ces deux machines est de l'ordre de sept). Actuellement, cette méthode n'est donc intéressante que pour des résolutions supérieures à 1024×1024.

3.4.4 Perspectives

On peut proposer plusieurs modifications à ce schéma :

1. Les dérivations du Jacobien peuvent être effectuées en base d'ondelettes : les étapes 2 et 3 deviennent alors :

   Étape 2 : On calcule à l'aide des opérateurs de dérivations en ondelettes $\vec{\nabla}\tilde{\psi}^n$ et $\vec{\nabla}\tilde{\omega}^n$ à partir de $\tilde{\psi}^n$ et $\tilde{\omega}^n$.

   Étape 3 : On effectue ensuite quatre transformées en ondelettes inverses (deux fonctions vectorielles comportant deux coordonnées chacunes) :

$$\overrightarrow{\nabla}\psi^n = FWT^{-1}(\overrightarrow{\nabla}\tilde{\psi}^n),\ \overrightarrow{\nabla}\omega^n = FWT^{-1}(\overrightarrow{\nabla}\tilde{\omega}^n),$$

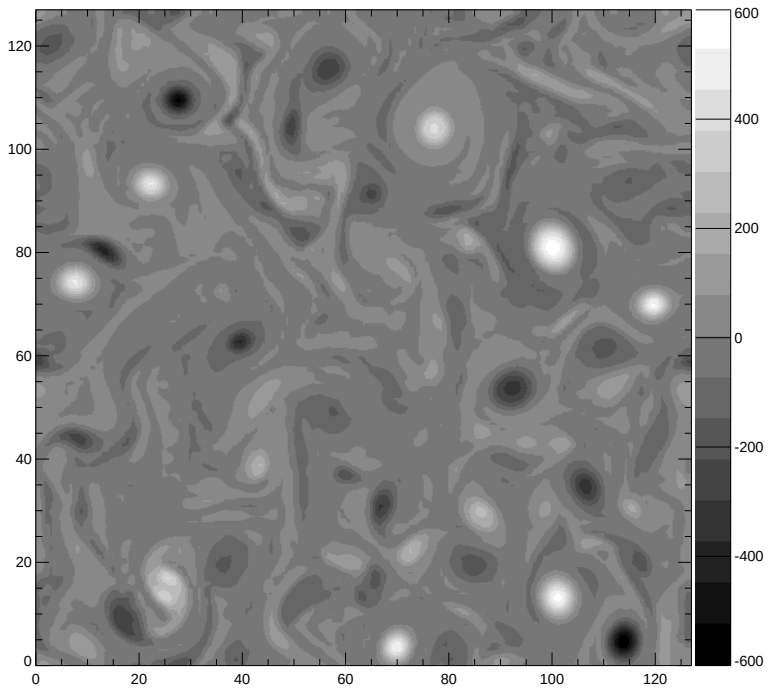
Étape 3bis : On calcule ensuite le Jacobien :

$$J(\psi^n,\omega^m) = \det(\overrightarrow{\nabla}\psi^{\mathrm{n}}, \overrightarrow{\nabla}\omega^{\mathrm{n}})$$

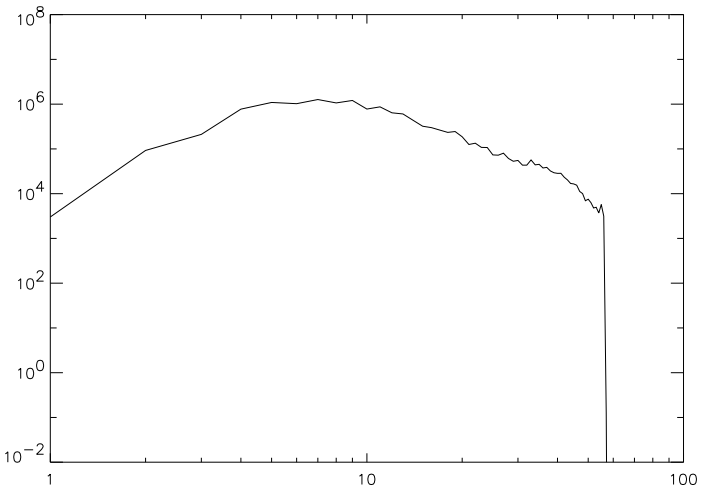
Cette méthode nécessite quatre transformées en ondelettes au lieu de deux initialement, il est donc moins coûteux d'effectuer cette opération dans l'espace physique, donc d'utiliser le schéma initial (le coût d'une transformée en ondelettes est deux fois supérieur à celui d'un calcul de dérivée par différences finies avec un schéma de même ordre que l'ondelette utilisée).

2. Il est également possible d'effectuer tous les calculs en bases d'ondelettes en utilisant des méthodes de Galerkin pour calculer le produit de deux fonctions en bases d'ondelettes. Ces méthodes ont été présentées dans [Beylkin91]. Des recherches sont en cours [Perrier et al.95] , mais actuellement, il est plus efficace d'effectuer cette opération dans l'espace physique, ce qui est également le cas pour les méthodes spectrales. On peut également signaler que le calcul du Jacobien, nécessite moins de 15% du temps total lors de la simulation en 128^2 présentée. Le changement de méthode ne peut donc pas permettre un gain de temps important, il peut par contre permettre une meilleure précision en espace, cette étape étant réalisée avec un schéma d'ordre inférieur.

3. Les ondelettes utilisées ici sont des ondelettes périodiques à support compact, elles permettent donc la résolution d'équations avec des conditions aux limites périodiques. Mais le développement récent d'ondelettes sur l'intervalle [Monasse et al.95] permet d'envisager la résolution des équations de Navier-Stokes avec toutes sortes de conditions aux limites.

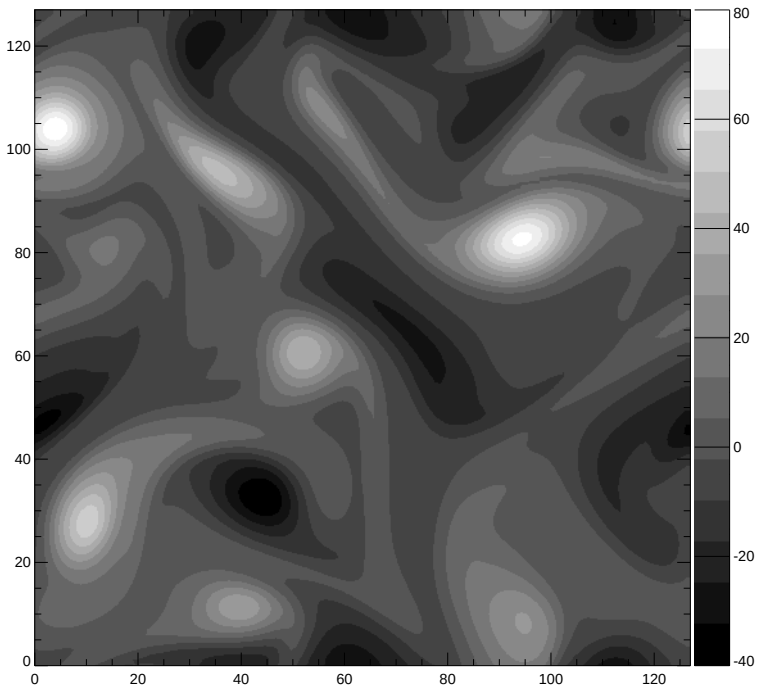


(a) Champ de vorticité

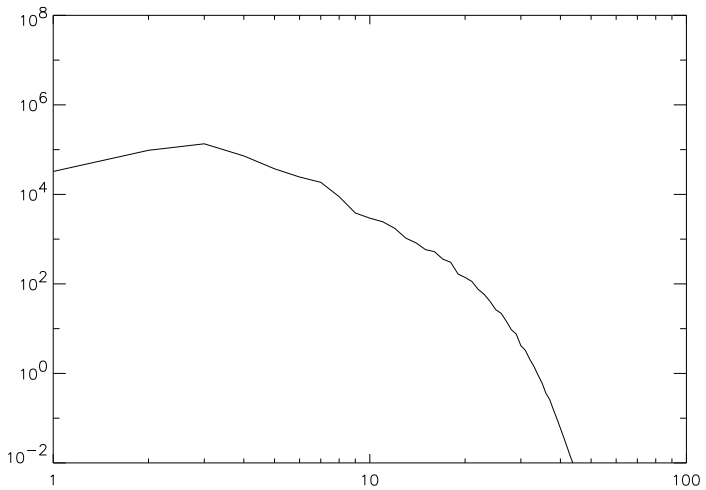


(b) Spectre d'enstrophie

FIG. 3.23 – *Condition initiale: initialfred*

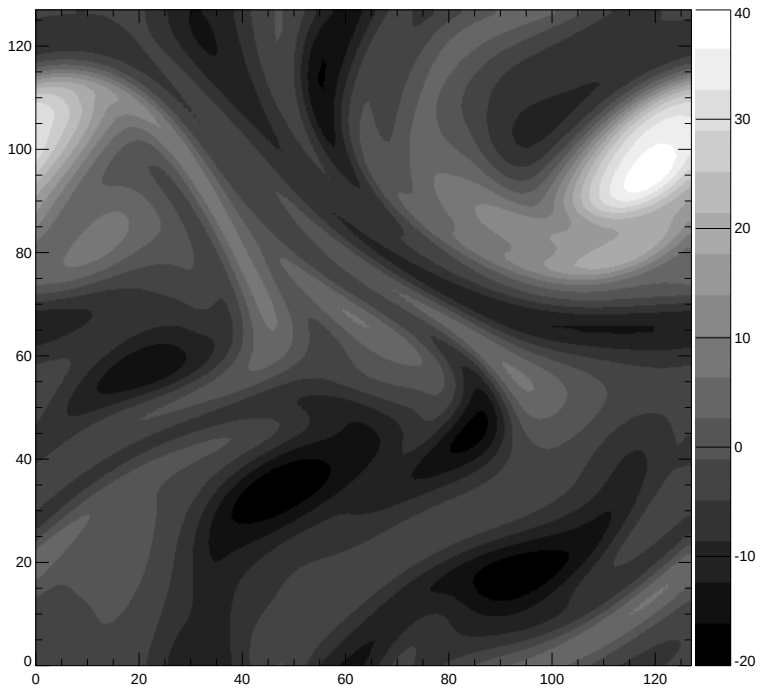


(a) Champ de vorticité

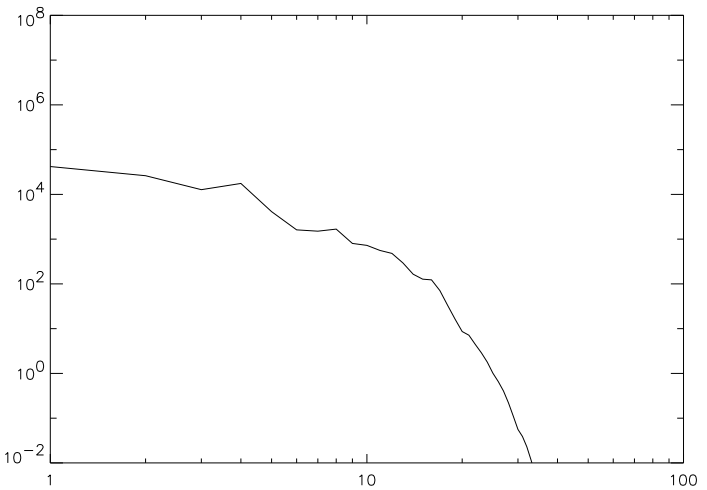


(b) Spectre d'enstrophie

FIG. 3.24 – *Champ au temps $t = 1$. (10 000 pas de temps)*



(a) Champ de vorticité



(b) Spectre d'enstrophie

FIG. 3.25 – *Champ au temps $t = 2$. (20 000 pas de temps)*

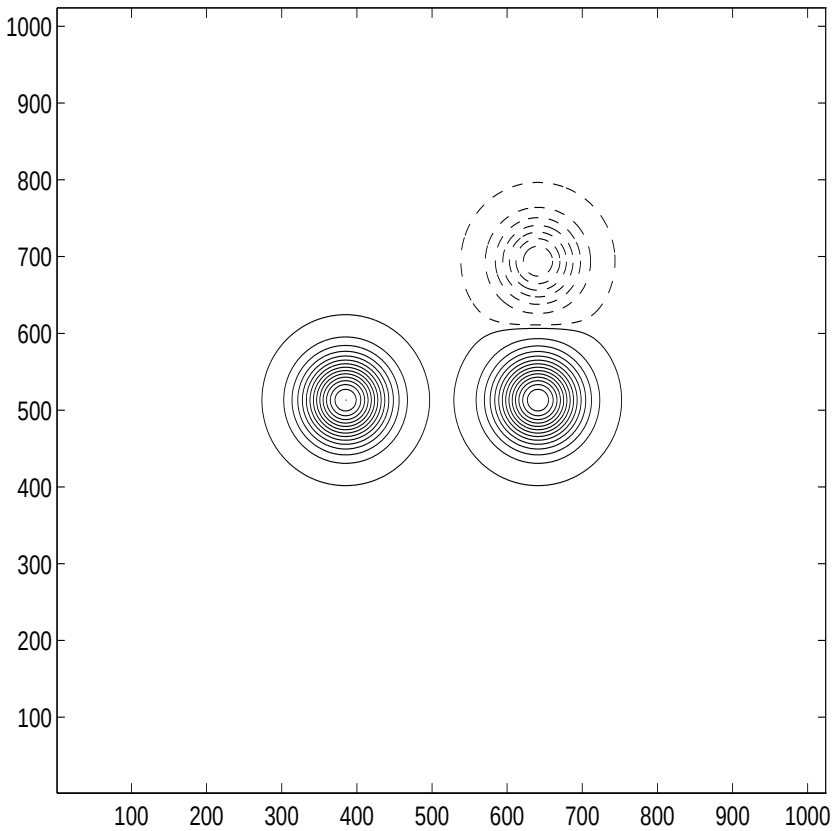
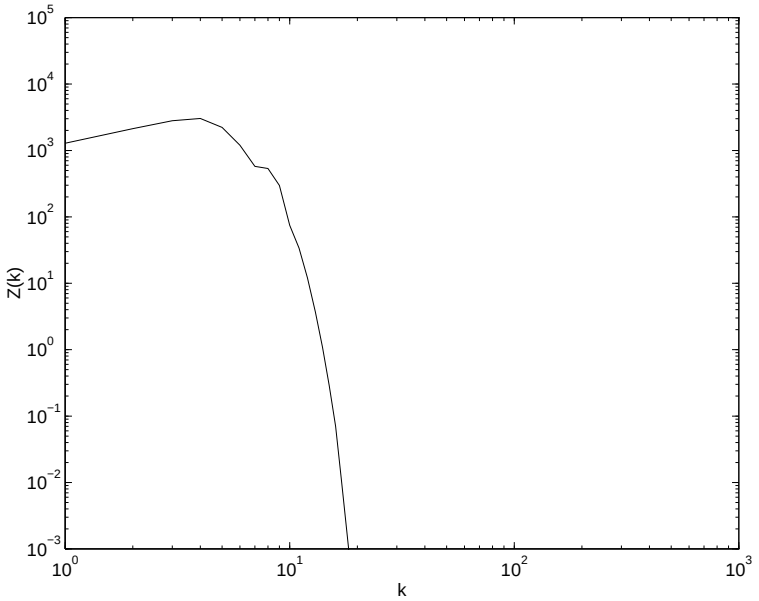
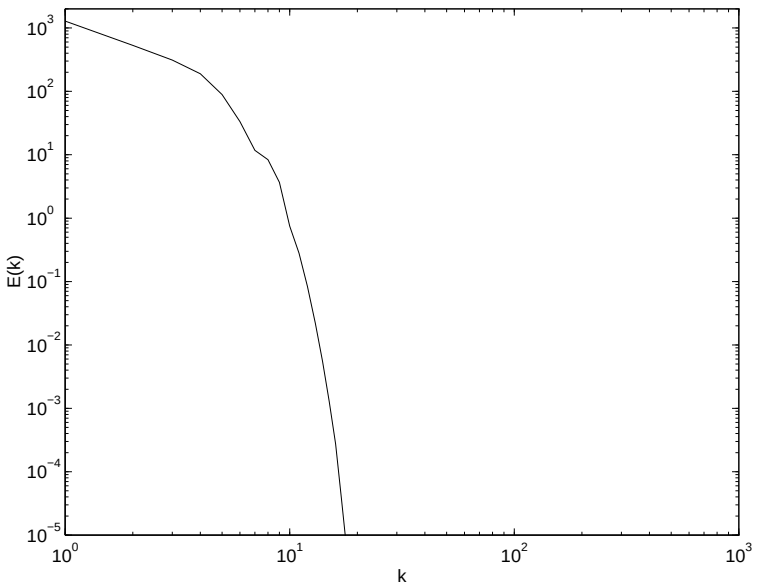


FIG. 3.26 – *Interaction de trois tourbillons : condition initiale*

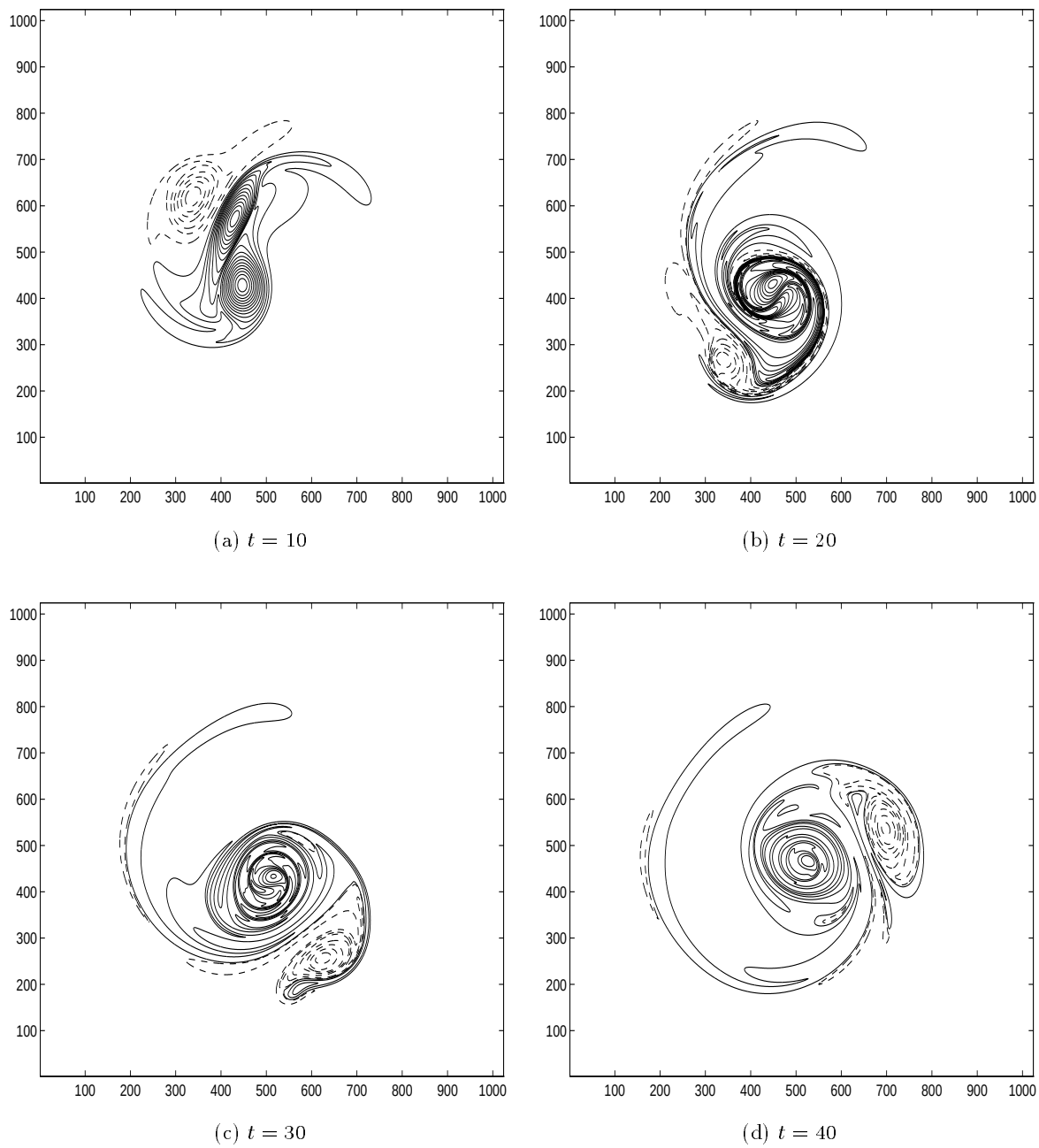


(a) Spectre d'enstrophie



(b) Spectre d'énergie

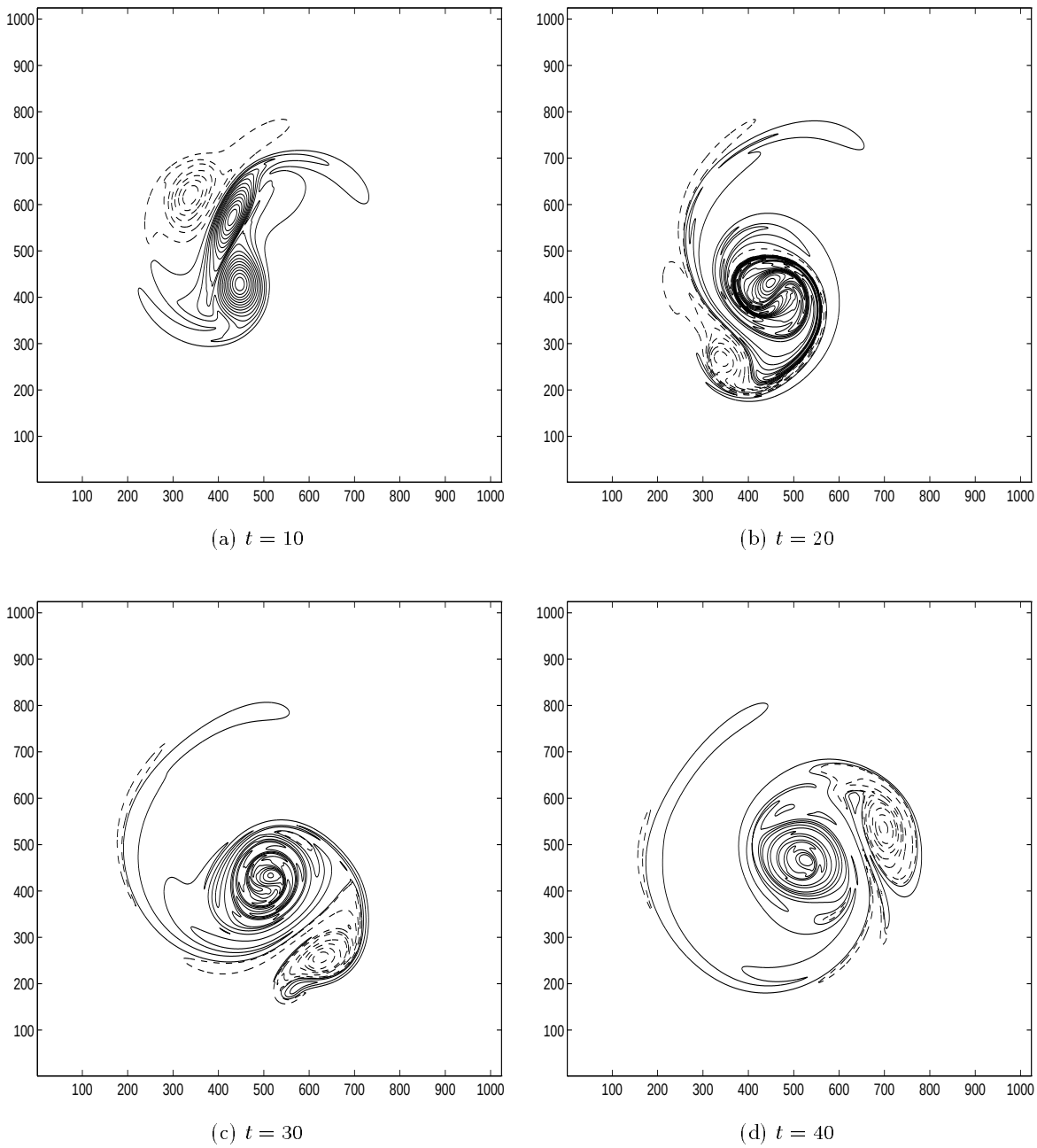
FIG. 3.27 – *Spectres d'enstrophie et d'énergie de la condition initiale*



(a) $t = 10$ (b) $t = 20$

(c) $t = 30$ (d) $t = 40$

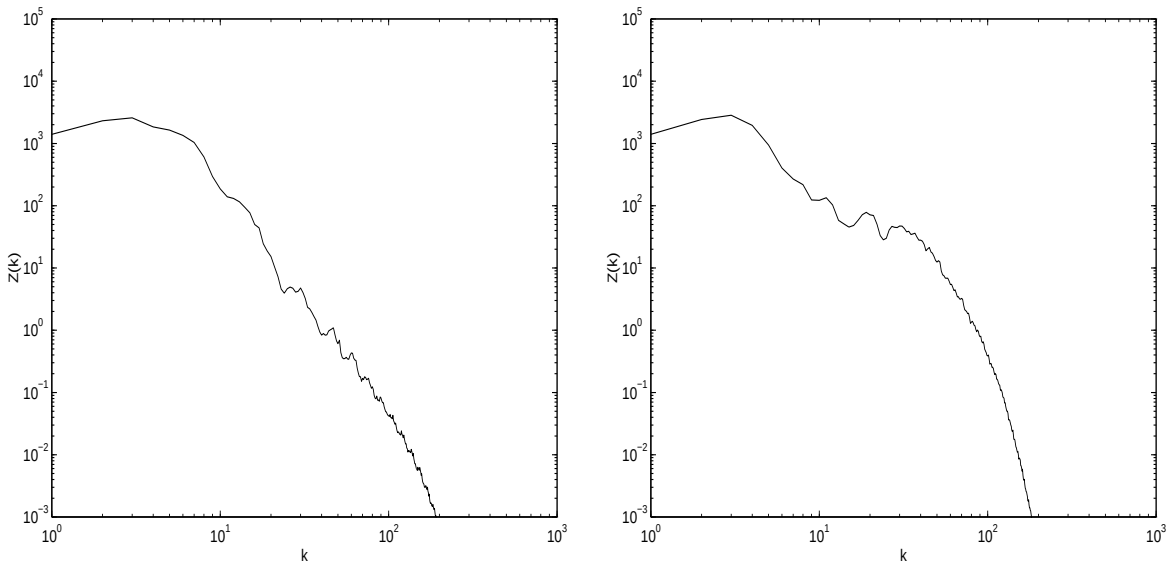
FIG. 3.28 – *Interaction de trois tourbillons : schéma de référence*



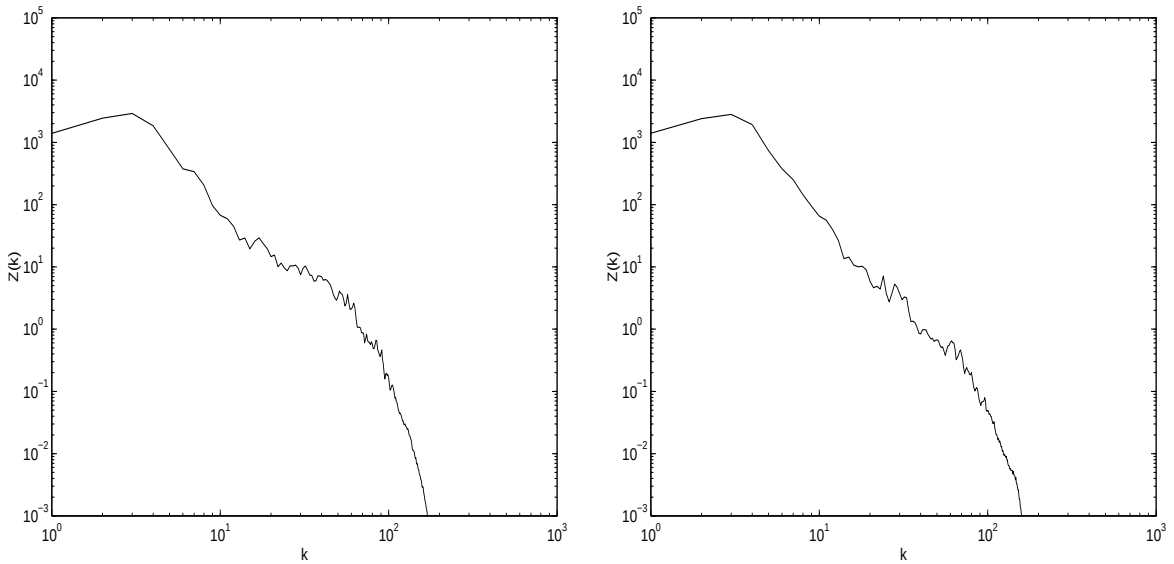
(a) $t = 10$ (b) $t = 20$

(c) $t = 30$ (d) $t = 40$

FIG. 3.29 – *Interaction de trois tourbillons : schéma en ondelettes*

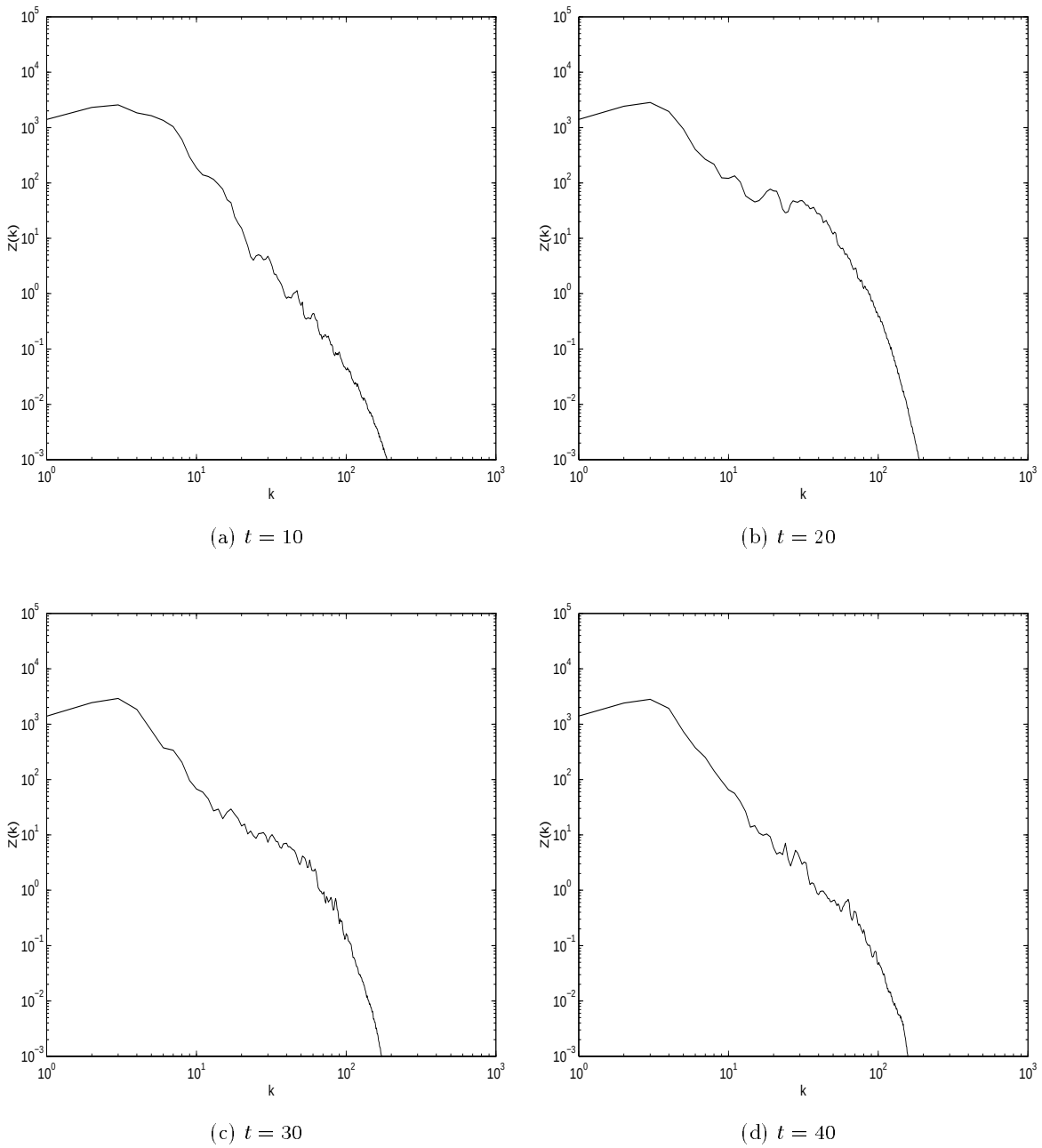


(a) $t = 10$ (b) $t = 20$



(c) $t = 30$ (d) $t = 40$

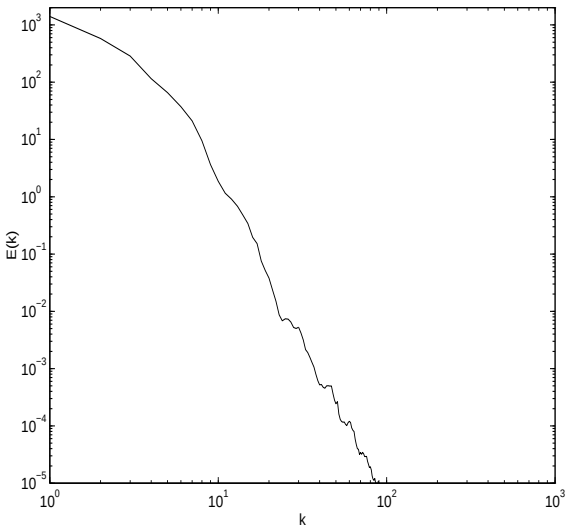
FIG. 3.30 – *Schéma de référence : spectres d'enstrophie*



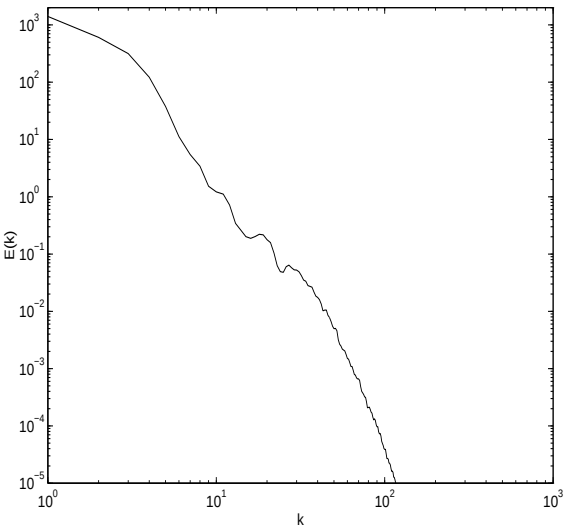
(a) $t = 10$ (b) $t = 20$

(c) $t = 30$ (d) $t = 40$

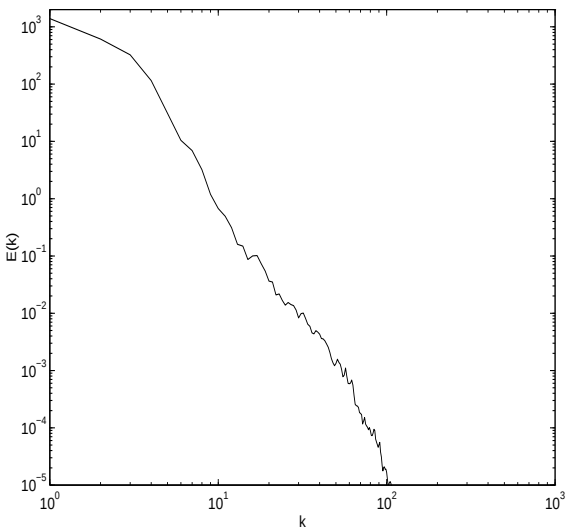
FIG. 3.31 – *Schéma en ondelettes : spectres d'enstrophie*



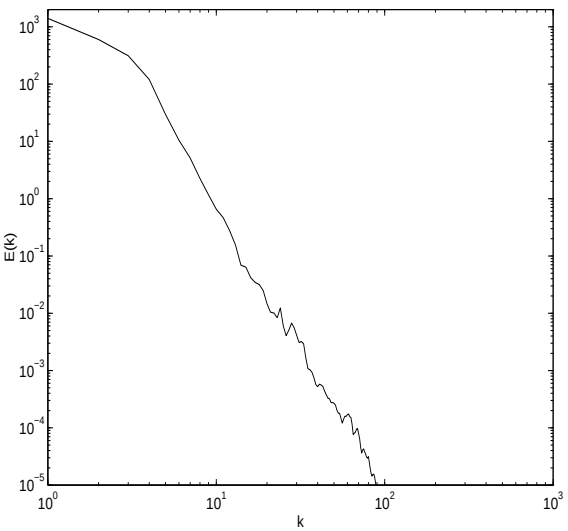
(a) $t = 10$



(b) $t = 20$

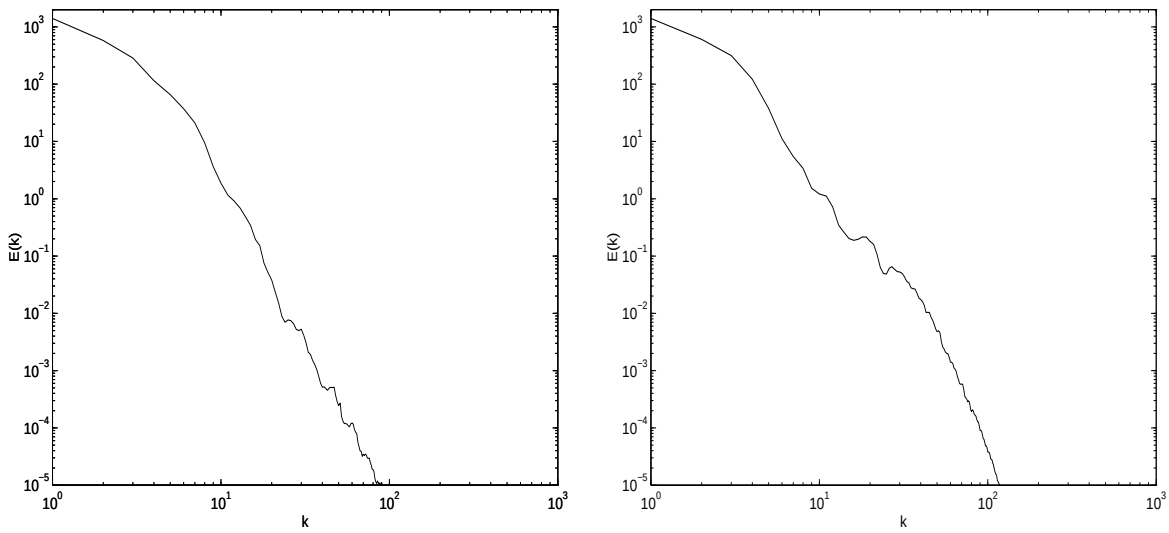


(c) $t = 30$

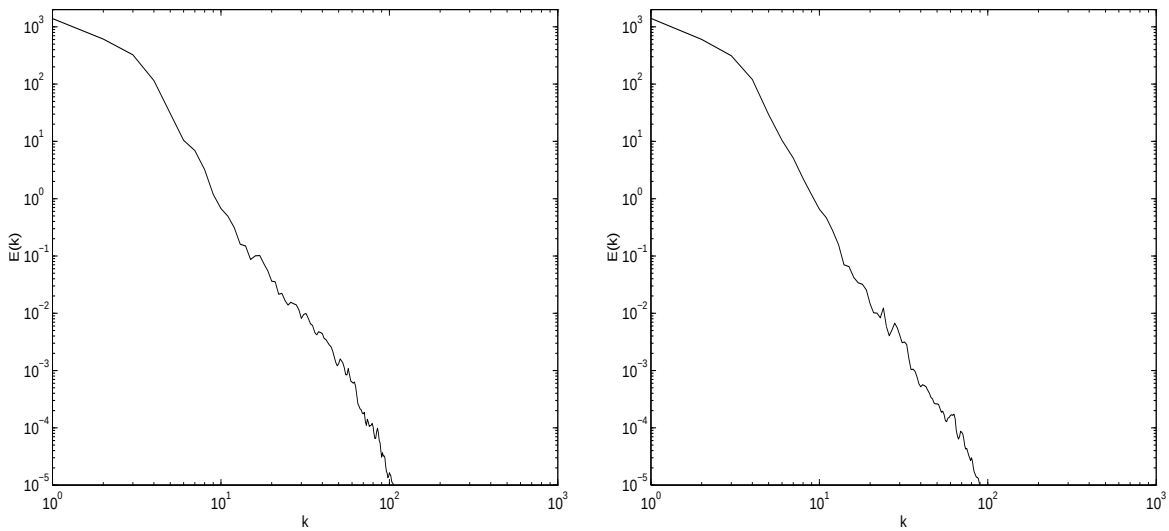


(d) $t = 40$

FIG. 3.32 – *Schéma de référence: spectres d'énergie*



(a) $t = 10$ (b) $t = 20$



(c) $t = 30$ (d) $t = 40$

FIG. 3.33 – *Schéma en ondelettes : spectres d'énergie*

Conclusion

Les ondelettes sont des fonctions oscillantes de moyenne nulle, qui essayent d'être le meilleur compromis entre les localisations spatiale et fréquencielle. La représentation dans une base d'ondelettes d'une fonction est donc un intermédiaire entre sa représentation dans l'espace physique et sa décomposition en modes de Fourier. Cette propriété permet de représenter de manière économique, par ses coefficients d'ondelettes, une fonction comportant des singularités isolées. Le noyau de Green d'un opérateur de Calderón-Zygmund est une fonction régulière, pouvant présenter une singularité sur la droite $y = x$ et décroissant à partir de cette diagonale ; les opérateurs de Calderón-Zygmund sont représentés efficacement par leurs coefficients d'ondelettes. Pour les opérateurs, il existe deux types de décomposition : la décomposition standard (décomposition sur une base tensorielle d'ondelettes) et la décomposition non standard (analyse multirésolution 2D de la matrice de l'opérateur).

La bonne représentation en ondelettes des fonctions et des opérateurs permet d'envisager de calculer rapidement l'action d'un opérateur sur un champ en se plaçant dans une base d'ondelettes. Dans cette perspective, nous développons un stockage et des algorithmes adaptés aux matrices creuses (utilisables quelle que soit la structure de la matrice), permettant d'effectuer rapidement des produits matrice-vecteur et des produits matrice-matrice. Le coût de ces opérations (initialement respectivement de N^2 et N^3 opérations, N étant la dimension du problème) est alors de $\frac{N_B N_X}{N}$ opérations (calcul du produit BX où N_B est le nombre d'éléments non nuls de B, N_X le nombre d'éléments non nuls de X et N le nombre total d'éléments de X).

Nous développons ensuite des algorithmes optimisés (au niveau du stockage et du temps de calcul) dans le cas où les opérateurs à décomposer sont invariants par translation (donc représentés par des matrices circulantes) : nous avons alors à stocker une matrice à la fois creuse et circulante que nous multiplions ensuite par un vecteur également creux. Ces algorithmes peuvent être employés dans la résolution des EDP à coefficients constants et permettent de diminuer considérablement le temps de calcul de la décomposition d'un opérateur en bases d'ondelettes et de réduire son stockage à quelques éléments.

Nous avons comparé à la section 3.1 les méthodes standard et non standard en les appliquant à la résolution de deux problèmes d'évolution : l'équation de la chaleur et l'équation de transport. Ces équations sont discrétisées par un schéma implicite en temps

et un schéma au différences finies en espace (on peut noter que les méthodes de Galerkin sur les ondelettes à support compact sont de type différences finies). Nous avons étudié l'effet des différents paramètres, ce qui permet de définir une stratégie pour les choisir. Les résultats numériques donnent pour les deux équations des temps de calcul plus faibles pour la méthode standard, malgré une représentation un peu meilleure des opérateurs par la décomposition non standard. Il est donc plus intéressant d'utiliser la méthode standard pour résoudre des problèmes d'évolution.

Pour la résolution d'EDP en dimension deux, nous nous appuyons sur une méthode de direction alternée (factorisation de l'opérateur de la chaleur), ce qui nous permet d'utiliser les méthodes développées en dimension un. Ceci a nécessité l'implémentation du calcul du produit d'une matrice creuse par la transposée d'une autre matrice creuse, le stockage utilisé devant donc être efficace dans les deux situations (produit AX et produit $X\ {}^tA$, en ne stockant que les matrices creuses A et X). Les résultats numériques obtenus sur l'équation de la chaleur, par comparaison avec la solution exacte, permettent de valider l'utilisation de la méthode des directions alternées et montrent que cette méthode s'applique efficacement à la résolution d'EDP de dimension supérieure ou égale à deux.

En nous appuyant sur le fait que les puissances itérées des opérateurs de Calderón-Zygmund sont également creuses en bases d'ondelettes, nous présentons une méthode itérative de résolution de l'équation de Poisson : il suffit de quelques itérations pour atteindre la convergence.

La combinaison de la méthode mise en oeuvre en dimension un et du schéma de directions alternées permet donc de résoudre des EDP à coefficients constants en dimension quelconque.

Nous résolvons ensuite une équation non linéaire : l'équation de Navier-Stokes bidimensionnelle incompressible dans sa formulation vorticité / fonction de courant. Le terme non linéaire est calculé par une méthode de collocation. Le nombre de degrés de liberté nécessaire à ce type de simulation est nettement réduit par l'utilisation des méthodes d'ondelettes, ce qui permet d'envisager la résolution de problèmes avec des résolutions supérieures à celles possibles avec les méthodes classiques. Ceci démontre que ces méthodes sont également applicables au calcul de la solution d'équations aux dérivées partielles non linéaires.

En ce qui concerne le terme non linéaire, il est également possible d'utiliser des méthodes de Galerkin permettant de bénéficier pleinement la réduction du nombre de degrés de liberté, mais celles-ci nécessitent d'être optimisées afin d'être compétitives avec les méthodes de collocation pour ce calcul.

Les ondelettes utilisées sont des ondelettes périodiques à support compact, les EDP résolues ont donc des conditions aux limites périodiques. Mais le développement récent d'ondelettes sur l'intervalle [Monasse et al.95] permet d'envisager la résolution d'EDP avec toutes sortes de conditions aux limites.

Il est intéressant de noter que ces méthodes peuvent également s'appliquer à tout problème nécessitant le calcul de produits de grandes matrices pleines comme ceux intervenant dans les équations intégrales issues par exemple de l'équation d'Helmoltz.

Annexe A

A.1 Calcul de l'inverse d'une matrice circulante

Soit $A = (a_{i-j})_{i,j}$ une matrice circulante et, B son inverse. Alors B est également circulante. Posons $B = (b_{i-j})_{i,j}$. Les coefficients b_j vont s'exprimer simplement en fonction des coefficients a en utilisant la transformation de Fourier discrète :

$$\widehat{a}_k = \frac{1}{N} \sum_{j=0}^{N-1} e^{i\frac{-2\pi jk}{N}} a_j \tag{A.1}$$

$$\text{Alors } \widehat{b}_k = \begin{cases} \frac{1}{\widehat{a}_k} & \text{si} \quad a_k \neq 0 \\ 0 & \text{si} \quad a_k = 0 \end{cases} \tag{A.2}$$

$$\text{Soit encore } b_j = \sum_{k=0}^{N-1} \frac{1}{\widehat{a}_k} e^{i\frac{2\pi jk}{N}} \tag{A.3}$$

Bibliographie

[Abgrall et al.94] R. Abgrall et A. Harten. – *Multiresolution representation in unstructured meshes: I.Preliminariy Report.* – Rapport technique n° 2342, INRIA, 9 1994.

[Alpert et al.89] B. Alpert et V. Rokhlin. – A fast algorithm for the evaluation of legendre expansions. *Yale University Technical Report*, n° YALEU/DCS/RR-496, 1989.

[Amaratunga et al.93] K. Amaratunga et J. Williams. – Wavelet bases Green's function approach to 2D PDEs. *Engineering Computations*, vol. 10, 93, pp. 349–367.

[Arakawa88] A. Arakawa. – Finite-difference methods in climate modeling, part I. *Physically-Based Modelling and Simulation of Climate and Climatic Change*, éd. par M.E. Schlesinger, pp. 79–168. – Kluwer Academic Publishers, 1988.

[Babiano et al.87] A. Babiano, C. Basdevant, B. Legras et R. Sadourny. – Vorticity and passive-scalar dynamics in two-dimensional turbulence. *J. Fluid Mech.*, vol. 183, 1987, pp. 379–397.

[Bacry et al.92] E. Bacry, S. Mallat et G. Papanicolaou. – A wavelet based space-time adaptive numerical method for partial differential equations. *Mathematical Modelling and Numerical Analysis*, vol. 26, 1992, pp. 793–834.

[Basdevant et al.81] C. Basdevant, B. Legras, R. Sadourny et M. Bélan. – A study of barotropic model flows : intermittency waves and predictability. *J. Atmos. Sci.*, vol. 38, 1981, pp. 2305–2326.

[Basdevant et al.90] C. Basdevant, M. Holschneider et V. Perrier. – Méthode des ondelettes mobiles. *C.R. Acad. Sci. Paris, Série I*, vol. 310, 1990, pp. 647–652.

[Battle93] G. Battle. – Klein-Gordon propagation of Daubechies wavelets. *J. Math. Phys.*, vol. 34, n° 3, 3 1993, pp. 1095–1109.

[Benzi et al.84] R. Benzi, G. Paladin, G. Parisi et A. Vulpiani. – Power spectra in two-dimensional turbulence. *J. Phys. A*, 1984.

[Bertoluzza et al.93] S. Bertoluzza et G. Naldi. – *Wavelet collocation method for Partial Differential Equations.* – Rapport technique n° 887, I.A.N., 1993.

[Bertoluzza et al.94a] S. Bertoluzza, Y. Maday et J.C. Ravel. – A dynamically adaptive wavelet method for solving partial differential equations. *Comput. Methods Appl. Mech. and Engrg.*, vol. 116, 1994, pp. 293–299.

[Bertoluzza et al.94b] S. Bertoluzza, G. Naldi et J.C. Ravel. – Wavelet methods for the numerical solution of boundary value problems on the interval. *Wavelets: Theory Algorithms and Applications*, éd. par L. Montefusco et Puccio. – C.K. Chui, Academic Press, 1994.

[Bertoluzza95a] S. Bertoluzza. – Adaptive wavelet collocation for the solution of steady-state equations. *SPIE*, 1995. – to appear.

[Bertoluzza95b] S. Bertoluzza. – Transport theory and statistical physics. 1995. – to appear.

[Beylkin et al.91] G. Beylkin, R. R. Coifman et V. Rokhlin. – Fast wavelet transforms and numerical algorithms I. *Comm. Pure and Appl. Math.*, vol. 44, 1991, pp. 141–183.

[Beylkin et al.93] G. Beylkin, R. R Coifman et V. Rokhlin. – Fast wavelet transforms and numerical algorithms II. – 1993. preprint.

[Beylkin91] G. Beylkin. – Wavelets, multiresolution analysis and fast numerical algorithms. – 1991. INRIA lectures.

[Beylkin92] G. Beylkin. – On the representation of operators in bases of compactly supported wavelets. *SIAM J. on Numerical Analysis*, vol. 29, n° 6, 1992, pp. 1716–1740.

[Beylkin93] G. Beylkin. – On wavelet-based algorithms for solving differential equations. – 1993. Soumis à CRC Press.

[Charton et al.95a] Ph. Charton et V. Perrier. – Factorisation sur bases d'ondelettes du noyau de la chaleur et algorithmes matriciels rapides associés. *C.R. Acad. Sci. Paris, Série I*, vol. 320, 1995, pp. 1013–1018.

[Charton et al.95b] Ph. Charton et V. Perrier. – Produits rapides matrice-vecteur en bases d'ondelettes : application à la résolution numérique d'équations aux dérivées partielles. M^2AN, vol. 29, n° 6, 1995, pp. 701–747.

[Cohen et al.92] A. Cohen, I. Daubechies et J.C. Feauveau. – Biorthogonal bases of compactly supported wavelets. *Comm. Pure and Appl. Math.*, vol. 45, 1992, pp. 485–560.

[Cohen et al.93] A. Cohen, I. Daubechies et P. Vial. – Wavelets on the interval and fast wavelet transforms. *App. Comp. Harmonic Analysis*, vol. 1, n° 1, 1993, pp. 54–81.

[Dahlke et al.94] S. Dahlke et I. Weinreich. – Wavelet bases adapted to pseudodifferential operators. *App. Comp. Harmonic Analysis*, vol. 1, 1994, pp. 267–283.

[Dahmen et al.92] W. Dahmen, S. Prößdorf et R. Schneider. – *Wavelet approximation methods for pseudodifferential equations I: stability and convergence.* – Rapport technique n° 77, RWTH Aachen, Institut für geometrie und praktische mathematik, 8 1992.

[Dahmen et al.93a] W. Dahmen, S. Prößdorf et R. Schneider. – *Wavelet approximation methods for pseudodifferential equations II: matrix compression and fast solution.* – Rapport technique n° 82, RWTH Aachen, Institut für geometrie und praktische mathematik, 2 1993.

[Dahmen et al.93b] W. Dahmen et C.A. Micchelli. – Using the refinement equation for evaluating integrals of wavelets. *SIAM J. on Numerical Analysis*, vol. 30, 1993, pp. 507–537.

[Dahmen et al.94] W. Dahmen, S. Prößdorf et R. Schneider. – Multiscale methods for pseudodifferential equations. *Recent Advances in Wavelet Analysis*, éd. par L.L. Schumaker et G. Webb, pp. 191–235. – Academic Press, 1994.

[Dahmen et al.95] W. Dahmen, A. Kunoth et K. Urban. – *A wavelet-Galerkin method for the Stokes-equations.* – Rapport technique n° 111, Aachen, I.G.P.M., 1995.

[Daubechies88] I. Daubechies. – Orthonormal bases of compactly supported wavelets. *Comm. Pure and Appl. Math.*, vol. 41, 1988, pp. 909–996.

[Daubechies92] I. Daubechies. – *Ten Lectures on Wavelets.* – SIAM, 1992, *CBMS-NSF Series in Applied Mathematics.*

[De Vore et al.92] R.A. De Vore, B. Jawerth et V. Popov. – Compression of wavelet decompositions. *American Journal of Mathematics*, vol. 114, 1992, pp. 737–785.

[Deschamps et al.85] V. Deschamps, P. Loisel et Y. Morchoine. – Recent development in inhomogeneous turbulence numerical simulation. 7 ^{e}Colloque International sur les méthodes Numériques de calcul Sc. et Tech. – Versailles, décembre 1985.

[Donoho93] D.L. Donoho. – Unconditionnal bases are optimal bases for data compression and statistical estimation. *App. Comp. Harmonic Analysis*, vol. 1, n° 1, 1993, pp. 100–115.

[Dorobantu et al.95] M. Dorobantu et B. Engquist. – *Wavelet-based acceleration of Jacobi iterations.* – Rapport technique, Dpt of Num. Anal., Royal Institute of Technology, Stockholm, 1995.

[Dorobantu92] M. Dorobantu. – *Wavelet based algorithms for one-dimensional parabolic equations.* – Rapport technique n° 9214, Dpt of Num. Anal., Royal Institute of Technology, Stockholm, 9 1992. Chaleur 1D explicite, calcul des puissances iterees, etude de stabilite du schema puissance iteree plus troncature.

[Dorobantu94] M. Dorobantu. – *Potential integral equations of the 2d Laplace operator in wavelet basis.* – Rapport technique, Dpt of Num. Anal., Royal Institute of Technology, Stockholm, 1994.

[Dorobantu95] M. Dorobantu. – *Wavelet-based algorithms for fast PDE solvers.* – Thèse de PhD, Stockholm University, Sep 1995.

[Elliott et al.95] F. Elliott et A. Majda. – A new algorithm whith plane waves and wavelets for random velocity fields withmany spatial scales. *J. of Comp. Physics*, vol. 117, 1995, pp. 146–162.

[Engquist et al.92] B. Engquist, S. Osher et S. Zhong. – Fast wavelet based algorithms for linear evolution equations. *ICASE Report*, n° 14, 1992.

[Farge et al.92] M. Farge, E. Goirand, Y. Meyer, F. Pascal et M. V. Wickerhauser. – Improved predictability of two-dimensional turbulent flows using wavelet packet compression. *Fluid Dynamics Research*, vol. 10, 1992, pp. 229–250.

[Farge92] M. Farge. – Wavelet transforms and their applications to turbulence. *Annal Review of Fluid Mechanics*, 1992, pp. 395–457.

[Fournier95] A. Fournier. – Wavelet representation of lower-atmospheric long nonlinear wave dynamics, governed by the Benjamin-Davis-ono-Burgers equation. – 1995. To appear in proceedings of Wavelet Applications for Dual Use.

[Fröhlich et al.94] J. Fröhlich et K. Schneider. – An adaptive wavelet galerkin algorithm for one and two dimensional flame computation. *Eur. J. Mech. B/Fluids*, vol. 13, n° 4, 1994, pp. 439–471.

[Fröhlich et al.95] J. Fröhlich et K. Schneider. – A fast algorithm for lacunary wavelet bases related to the solution of PDEs. – 1995. preprint, soumis au C.R.A.S.

[Gagnon et al.94] L. Gagnon et J. M. Lina. – *Symmetric Daubechies' wavelets and numerical solution of NLS equations.* – Rapport technique n° 16, Lab. de Phys. Nucléaire, Univ. Montréal, 1994.

[Grossmann et al.84] A. Grossmann et J. Morlet. – Decomposition of hardy functions into square integrable wavelets of constant shape. *SIAM J. on Math. Analysis*, vol. 15, 1984, pp. 723–736.

[Harten et al.92] A. Harten et I. Yad-Shalom. – *Fast multiresolution algorithms for matrix-vector multiplication.* – Rapport technique n° 92-55, ICASE, 10 1992.

[Harten94a] A. Harten. – Adaptative multiresolution schemes for shock computations. *J. of Comput. Phys.*, vol. 115, 1994, pp. 319–338.

[Harten94b] A. Harten. – Multiresolution representation and numerical algorithms: a brief review. *ICASE Report*, n° 59, 1994.

[Holschneider et al.91] M. Holschneider et P. Tchamitchian. – Pointwise regularity of Riemann's 'nowhere differentiable' function. *Inventiones Mathematicae*, vol. 105, 1991, pp. 157–175.

[Jaffard et al.89] S. Jaffard et Y. Meyer. – Bases d'ondelettes dans des ouverts de $\mathbb{R}^n$. *J. Math. Pures Appl.*, vol. 68, 1989, pp. 95–108.

[Jaffard et al.95] S. Jaffard et Y. Meyer. – Wavelet methods for pointwise regularity and local oscillations of functions. – 1995. à paraitre dans : Memoirs of the A.M.S.

[Jaffard89] S. Jaffard. – Exposants de hölder en des points donnés et coefficients d'ondelettes. *C.R. Acad. Sci. Paris, Série I*, vol. 308, 1989, pp. 79–81.

[Jaffard92] S. Jaffard. – Wavelet methods for fast resolution of elliptic problems. *SIAM J. on Numerical Analysis*, vol. 29, n° 4, 1992, pp. 965–986.

[Jameson93] L. Jameson. – *On the Daubechies-based wavelet differentiation matrix.* – Rapport technique n° 93–80, ICASE, 1993.

[Jameson94] L. Jameson. – *On the wavelet optimized finite difference method.* – Rapport technique n° 94-9, ICASE, 1994.

[Kleemann95] B. Kleemann. – On the solution of the helmholtz and laplace equations using wavelets. – 1995. preprint Berlin.

[Landau et al.59] L.D. Landau et E.M. Lifshitz. – *Physique théorique 6 : Mécanique des fluides.* – Editions MIR, 1959.

[Latto et al.90] A. Latto et E. Tenebaum. – Compactly support wavelets and numerical solution of Burgers' equation. *C.R. Acad. Sci. Paris, Série I*, vol. 311, 1990, pp. 903–909.

[Lazaar et al.94a] S. Lazaar, J. Liandrat et P. Tchamitchian. – Algorithme à base d'ondelettes pour la résolution numérique d'équations aux dérivées partielles à coefficients variables. *C.R. Acad. Sci. Paris, Série I*, vol. 319, 1994, pp. 1101–1107.

[Lazaar et al.94b] S. Lazaar, P.J. Ponenti, J. Liandrat et P. Tchamitchian. – Wavelet algorithms for numerical resolution of partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, vol. 116, 1994, pp. 309–314.

[Lemarié95] P. G. Lemarié. – Pseudo-invariance par translations et dilatations dyadiques et transformées en ondelettes rapides. – 1995. Journée ondelettes et analyse numerique, Orsay, 8 juin.

[Liandrat et al.90a] J. Liandrat et F. Moret-Bailly. – The wavelet transform: some applications to fluid dynamics and turbulence. *Eur. J. Mech.*, vol. 9, 1990, pp. 1–19.

[Liandrat et al.90b] J. Liandrat et P. Tchamitchian. – *Resolution of the 1D regularized Burgers equation using spatial wavelet approximation.* – Rapport technique n° 90-83, ICASE, 1990.

[Liandrat et al.91] J. Liandrat, V. Perrier et P. Tchamitchian. – Numerical resolution of nonlinear partial differential equations using the wavelet approach. *Wavelet and their applications*, éd. par Ruskai et al., pp. 227–238. – Jones and Barlet, 1991.

[Maday et al.91] Y. Maday, V. Perrier et J. C. Ravel. – Adaptativité dynamique sur bases d'ondelettes pour l'approximation d'équations aux dérivées partielles. *C.R. Acad. Sci. Paris, Série I*, vol. 312, 1991, pp. 405–410.

[Maday et al.92] Y. Maday et J. C. Ravel. – Adaptativité par ondelettes : conditions aux limites et dimensions supérieures. *C.R. Acad. Sci. Paris, Série I*, vol. 315, 1992, pp. 85–90.

[Mallat89] S. Mallat. – Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$. *IEEE, transactions on PAMI*, vol. 2, n° 7, 1989.

[Mane87] L. Mane. – *Parallélisation sur un système multiprocésseur d'une méthode de résolution des équations de Navier-Stokes instationnaires à grands nombres de Reynolds.* – Thèse de doctorat, Université Paris 6, 1987.

[Meyer90] Y. Meyer. – *Ondelettes et Opérateurs I et II.* – Hermann, 1990.

[Meyer92] Y. Meyer. – Ondelettes sur l'intervalle. *Revista Matemàtica Iberoamericana*, vol. 7, 1992, pp. 115–133.

[Monasse et al.95] P. Monasse et V. Perrier. – Ondelettes sur l'intervalle pour la prise en compte de conditions aux limites. *C.R. Acad. Sci. Paris, Série I*, vol. 321, 1995, pp. 1163–1169.

[Nikolskii05] S. M. Nikolskii. – *Analysis III, Spaces of differentiable functions.* – Springer-Verlag, 1905, *Encyclopaedia of mathematical sciences.*

[Pathria et al.] D. Pathria et G. Karniadakis. – Wavelet-aided spectral element flow simulations. – preprint.

[Peaceman et al.55] D.W. Peaceman et H.H. Rachford. – The numerical solution of parabolic and elliptic differential equations. *J. Soc. Ind. Appl. Math.*, vol. 3, 1955, pp. 28–42.

[Perrier et al.89] V. Perrier et C. Basdevant. – La décomposition en ondelettes périodiques, un outil pour l'analyse de champs inhomogènes. Théorie et algorithmes. *La Recherche Aérospatiale*, vol. 3, 1989, pp. 53–67.

[Perrier et al.95] V. Perrier et M.V. Wickerhauser. – Multiplication of short wavelet series using connection coefficients. – 1995. preprint Saint Louis, USA.

[Ponenti94] P.J. Ponenti. – *Algorithmes en ondelettes pour la résolution d'équations aux dérivées partielles.* – Thèse de doctorat, Université d'Aix-Marseille I, 1994.

[Qian et al.93] S. Qian et J. Weiss. – Wavelets and the numerical solution of partial differential equations. *J. of Comp. Phys.*, vol. 106, 1993, pp. 155–175.

[Rathsfeld94] A. Rathsfeld. – A wavelet algorithm for the solution of the double layer potential equation over polygonal boundaries. – 1994. preprint Berlin.

[Santangelo et al.89] P. Santangelo, R. Benzi et B. Legras. – The generation of vortices in high-resolution, two-dimensional decaying turbulence and the influence of initial conditions on the breaking of self-similarity. *Phys. Fluids*, vol. 1, n° 6, 1989, pp. 1027–1034.

[Stein70] E.M. Stein. – *Singular Integrals and Differentiability Properties of Functions.* – Princeton, New Jersey, Princeton University Press, 1970.

[Urban94] K. Urban. – *A wavelet-Galerkin algorithm for the driven-cavity-Stokes-problem in two space dimensions.* – Rapport technique, R.W.T.H. Aachen, 1994.

[Wachpress62] E.L. Wachpress. – Optimum alterning-direction-implicit iteration parameters for a model problem. *J. Soc. Ind. Appl. Math.*, vol. 10, n° 2, 1962, pp. 339–350.

[Walden] J. Walden. – Spectral analysis of the differentiation operator in wavelet bases. – preprint.

[Yanenco68] N.N. Yanenco. – *Méthodes à pas fractionnaires.* – Armand Colin, 1968, *Collection Intersciences.*