A paraître dans M^2AN

Produits rapides matrice-vecteur en bases d'ondelettes : application à la résolution numérique d'équations aux dérivés partielles

Philippe CHARTON * Valérie PERRIER *

23 juin 1995

Résumé

Certains produits matrice-vecteur peuvent être effectués de manière rapide en utilisant la transformée en ondelettes. Nous comparons dans cet article deux méthodes différentes : la méthode standard, simple transformée en ondelettes du système et la décomposition BCR. Les algorithmes et la mise en œuvre de ces méthodes sont détaillés et optimisés dans le cas où la matrice est circulante.

Ces méthodes sont appliquées à la résolution numérique de l'équation de la chaleur et de l'équation de transport, unidimensionnelles avec conditions aux limites périodiques : des tests portant sur les temps de calcul sont présentés, ils démontrent le coût inférieur de la méthode standard.

Abstract

Some matrix-vector products can be efficiently computed using the wavelet transform. In this article, we compare two methods: the standard method, which simply wavelet transforms the system, and the BCR method. These algorithms and the practical implementation of the two methods are explained in detail and improvements are given for the shift-invariant operator case.

These methods are applied to the numerical solution of heat equation and advection equation for a periodic one-dimensional domain: computing time comparisons are given, demonstrating the superiority of the standard method in terms of computational cost.

1 Introduction

L'étude des écoulements turbulents bidimensionnels utilise intensivement la résolution numérique de l'équation de Navier-Stokes pour les fluides incompressibles. Celle-ci peut s'écrire (en l'absence de forçage) comme la simple advection-diffusion du tourbillon (ou vorticité). Pour des études de turbulence homogène on utilise des conditions aux limites périodiques dans les deux directions, les équations à discrétiser et intégrer numériquement sont alors les suivantes :

$$\begin{cases} \frac{\partial \omega}{\partial t} + \overrightarrow{u} \cdot \nabla\omega - \nu\Delta\omega = 0 \\ \omega(\overrightarrow{x}, t) \ 2\pi\text{-périodique en } \overrightarrow{x} \ , \ \overrightarrow{x} \in \mathbb{T}^2, t \geq 0 \\ \omega = rot\,\overrightarrow{u} \\ \omega(\overrightarrow{x}, 0) = \omega_0(\overrightarrow{x}) \end{cases} \tag{1}$$

*. Laboratoire d'Analyse, Géométrie et Applications, UA 742 Institut Galilée, Université Paris Nord, Av J.B. Clément 93430 Villetaneuse et Laboratoire de Météorologie Dynamique , E.N.S., 24 rue Lhomond, 75231 Paris cedex 05

La modélisation directe de cette équation utilise classiquement une méthode spectrale ou pseudo-spectrale [Basdevant et al.81], facilitée par l'emploi de la transformée de Fourier rapide (FFT). Le problème majeur lors de ces simulations réside dans le très grand nombre de paramètres à gérer : en dimension deux, celui-ci varie comme Re, où Re est le nombre de Reynolds de l'écoulement (inversement proportionnel à la viscosité ν). Re, dans les écoulements réalistes, peut atteindre des valeurs de plusieurs milliers en conditions industrielles et du million pour l'écoulement atmosphérique à grande échelle. On arrive très vite hors de portée des calculateurs actuels. La situation est encore plus défavorable pour les écoulements en dimension 3.

Un domaine de recherche consiste à étudier de nouvelles représentations pour la solution de l'équation qui soient mieux adaptées à la structure spatiale très intermittente de l'écoulement que ne le sont les méthodes spectrales ou les méthodes de différences finies.

En effet, analysé en espace et en temps, un écoulement turbulent bidimensionnel peut être schématisé comme une succession d'interactions non linéaires entre des « structures cohérentes » de tailles très différentes formées d'un ou plusieurs tourbillons. Il est clair que l'analyse classique de Fourier n'est pas un bon outil pour représenter des champs ayant une telle variabilité spatiale. Par contre un outil bien adapté pour décrire de telles structures (localisées à la fois en espace et en échelle) est la transformée en ondelettes [Farge92] : analysé sur une base d'ondelettes, un champ turbulent nécessite peu de degrés de libertés pour une description précise, comparativement à une analyse de Fourier, les coefficients significatifs de l'analyse en ondelettes se concentrant dans les zones présentant des petites structures. Un outil encore plus efficace pour la compression des champs est la transformée en paquets d'ondelettes [Farge et al.92], il s'avère toutefois d'une utilisation compliquée pour la résolution des EDP. D'autre part, les bases orthonormées d'ondelettes sont associées à des algorithmes rapides d'analyse et de synthèse analogues à la FFT (voire plus rapides) permettant de passer commodément des coefficients selon la base aux valeurs aux nœuds d'un maillage régulier.

Si l'application des ondelettes à la simulation numérique des EDP progresse lentement, cela tient à la difficulté pratique de leur mise en œuvre, elle-même liée au fait que contrairement à Fourier, les opérateurs de dérivation ne sont pas diagonaux en ondelettes. Toutefois, certains opérateurs pseudo-différentiels possèdent la propriété d'être presque diagonaux dans les bases d'ondelettes [Meyer90, Beylkin et al.91] : c'est le cas en particulier pour les opérateurs ∇ et $I - \Delta$, mais surtout pour les opérateurs intégraux comme le noyau de Green $(I - \Delta)^{-1}$ qui sont des opérateurs de Calderón-Zygmund (voir [Meyer90] pour la définition de ces opérateurs).

Les schémas actuels de décompositions en ondelettes pour la résolution des équations aux dérivées partielles sont essentiellement partagés en deux catégories : la première comprend la construction de bases biorthogonales d'ondelettes associées à des opérateurs elliptiques, ce qui réduit le système à de simples changements de bases, utilisant des algorithmes rapides en cascade [Liandrat et al.92, Dahlke et al.94, Lazaar et al.94].

La seconde catégorie concerne les méthodes de Galerkin classiques, conduisant à la résolution de systèmes linéaires : l'idée est alors de tirer parti de la « lacunarité » de la représentation en ondelettes de la solution et de l'opérateur. Plusieurs approches ont déjà été développées : la décomposition BCR [Beylkin et al.91] s'applique à des équations intégrales ; les équations elliptiques peuvent être résolues par une méthode iterative en utilisant un préconditionneur diagonal sur base d'ondelettes [Jaffard92]. Enfin, pour les équations d'évolution, les coefficients d'ondelettes non nuls de la solution peuvent être suivis dynamiquement au cours du temps [Maday et al.91], le pas de temps adapté en fonction de l'échelle [Bacry et al.91].

Notre article se situe dans la seconde catégorie : l'objectif de la présente étude est de comparer deux méthodes matricielles de résolution d'EDP utilisant la transformée en ondelettes, dans le but de résoudre les équations de Navier-Stokes bidimensionnelles incompressibles.

De façon classique, on considérera un schéma semi-implicite en temps ; implicite pour le terme de diffusion pour des raisons de stabilité, explicite pour le terme d'advection non-linéaire, soit symboliquement (l'objet n'est pas ici de discuter de l'ordre du schéma) :

$$\begin{cases} \omega(\overrightarrow{x}, n\Delta t) = \omega^n(x) \\ (I - \delta t\nu\Delta)\omega^{n+1} = \omega^n - \delta t(\overrightarrow{u}^n \cdot \nabla\omega^n) \end{cases} \tag{2}$$

Deux problèmes de natures différentes se présentent : la résolution du système linéaire dûe à l'implicitation en temps et le calcul du terme non-linéaire. Cette étude est consacrée au premier problème, le calcul du terme non linéaire sera discuté dans une publication ultérieure.

Notre point de vue est de tirer parti de la représentation lacunaire en base d'ondelettes des noyaux d'opérateurs intégraux : dans cette optique nous présenterons et comparerons deux méthodes basées sur l'analyse en ondelettes pour résoudre des systèmes linéaires du type $u = (I - \delta t\nu\Delta)^{-1}f$. L'étude est faite, dans un premier temps, en dimension un, afin de dégager la méthode la plus efficace et économique (tant du point de vue du nombre d'opérations, que du stockage de l'opérateur), pour effectuer dans un deuxième temps l'extension **réaliste** en dimension deux [Charton et al.95].

Dans la section 2 nous rappelons le principe de la construction des bases orthonormées d'ondelettes et des algorithmes d'analyse et de synthèse. Dans la section 3, nous détaillons la discrétisation pratique d'un problème modèle d'EDP continu en un système linéaire. Les différentes transformations du système linéaire sont décrites en partie 4, puis optimisées en partie 5, dans le cas où la matrice du système est circulante (ce qui correspond effectivement à l'opérateur $I - \nu\delta t\Delta$ de l'équation (2)).

La partie 6 donne les détails techniques de mise en œuvre de la résolution du système. Les résultats numériques, concernant deux problèmes test sont développés en partie 7. Une synthèse des différentes méthodes, en vue de l'extension en dimension deux, clôt cet article.

2 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'analyses multirésolutions [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.

2.1 Analyse multirésolution sur $\mathbb{R}$

Définition 1 *Une analyse multirésolution r-régulière 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 :*

$$\forall j \in \mathbb{Z}, \qquad V_j \subset V_{j+1} \tag{3}$$

$$\bigcap_{j\in\mathbb{Z}} V_j = \{0\} \quad et \quad \bigcup_{j\in\mathbb{Z}} V_j \text{ est dense dans } L^2(\mathbb{R}) \tag{4}$$

$$\forall f \in L^2(\mathbb{R})\ \forall j \in \mathbb{Z}, \qquad f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1} \tag{5}$$

$$\exists \varphi \in V_0 \text{ telle que} \{\varphi(x-k)\}_{k\in\mathbb{Z}} \text{ soit une base orthonormée de } V_0. \tag{6}$$

$$\varphi \text{ est dérivable jusqu'à l'ordre } r \text{ et ses dérivées sont à décroissance rapide.} \tag{7}$$

Remarques : De la propriété (6), on déduit : $\forall f \in L^2(\mathbb{R})\ \forall k \in \mathbb{Z}, f(x) \in V_0 \Leftrightarrow f(x-k) \in V_0$, et de (5) que $\{2^{\frac{j}{2}}\varphi(2^j x - k)\}_{k\in\mathbb{Z}}$ est une base orthonormée de V_j.

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

$$V_{j+1} = V_j \oplus W_j \tag{8}$$

On a donc la décomposition de $L^2(\mathbb{R})$ suivante :

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

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 (7) ainsi que la propriété suivante dite de « moments nuls » :

$$\int x^q \psi(x)\ dx = 0 \text{ pour } 0 \leq q \leq r. \tag{9}$$

Les propriétés (7) et (9) 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 [Jaffard89, Meyer90, Holschneider et al.91, De vore et al.92, Donoho93].

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 complexité théorique plus faible. Les détails de construction de ces bases d'ondelettes sont explicités dans [Meyer90] et [Daubechies92].

2.2 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. 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 (5) 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 :

$$\varphi(x) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_k\ \varphi(2x-k) \text{ avec } h_k = \int_{-\infty}^{+\infty} \varphi(x)\cdot\sqrt{2}\ \varphi(2x-k)\ dx \tag{10}$$

$$\psi(x) = \sqrt{2}\sum_{k\in\mathbb{Z}} g_k\ \varphi(2x-k) \text{ avec } g_k = \int_{-\infty}^{+\infty} \psi(x)\cdot\sqrt{2}\ \varphi(2x-k)\ dx \tag{11}$$

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

$$g_k = (-1)^k h_{1-k} \tag{12}$$

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

$$s_k^j = \int_{-\infty}^{+\infty} f(x)\varphi_k^j(x)dx \quad \text{et} \quad d_k^j = \int_{-\infty}^{+\infty} f(x)\psi_k^j(x)dx \tag{13}$$

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 décalle l'entrée de deux points à chaque étape) :

$$s_k^{j-1} = \sum_{n\in\mathbb{Z}} h_n s_{2k+n}^j \tag{14}$$

$$d_k^{j-1} = \sum_{n\in\mathbb{Z}} g_n s_{2k+n}^j \tag{15}$$

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 :

$$\begin{array}{ccccccc} (s_k^0)_{k=0,N-1} & \longrightarrow & (s_k^{-1})_{k=0,\frac{N}{2}-1} & \longrightarrow & (s_k^{-2})_{k=0,\frac{N}{4}-1} & \longrightarrow & (s_k^{-3})_{k=0,\frac{N}{8}-1} \quad \cdots \\ & \searrow & & \searrow & & \searrow & \\ & & (d_k^{-1})_{k=0,\frac{N}{2}-1} & & (d_k^{-2})_{k=0,\frac{N}{4}-1} & & (d_k^{-3})_{k=0,\frac{N}{8}-1} \quad \cdots \end{array} \tag{16}$$

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 partis 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 (8) est la suivante :

$$s_k^{j+1} = \sum_{n\in\mathbb{Z}} h_{k-2n}s_n^j + \sum_{n\in\mathbb{Z}} g_{k-2n}d_n^j \tag{17}$$

et l'on utilise l'algorithme pyramidal (16) 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 [Meyer90, Perrier et al.89]; 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.

2.3 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*

$$\forall j \in \mathbb{N}, \qquad V_j \subset V_{j+1} \tag{18}$$

$$\bigcup_{j\in\mathbb{N}} V_j \text{ est dense dans } L^2(\mathbb{T}) \tag{19}$$

$$V_0 = \{f = C^{te}\} \tag{20}$$

$$f(x) \in V_j \Rightarrow f(2x) \in V_{j+1} \tag{21}$$

$$f(x) \in V_{j+1} \Rightarrow f\left(\frac{x}{2}\right) + f\left(\frac{x}{2} + \frac{1}{2}\right) \in V_j. \tag{22}$$

$$\varphi \text{ est dérivable jusqu'à l'ordre } r \text{ et ses dérivées sont à décroissance rapide.} \tag{23}$$

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

$$\varphi_k^j(x) = \varphi_0^j(x - \frac{k}{2^j}) \tag{24}$$

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 $\mathbb{R}$ (14-15) en périodisant les filtres à chaque échelle 2^j :

$$s_i^{j-1} = \sum_{n=0}^{2^j-1} h_n^j s_{2i+n}^j \qquad \text{avec} \qquad h_n^j = \sum_{k\in\mathbb{Z}} h_{n+2^j k} \tag{25}$$

$$d_i^{j-1} = \sum_{n=0}^{2^j-1} g_n^j s_{2i+n}^j \qquad \text{avec} \qquad g_n^j = \sum_{k\in\mathbb{Z}} g_{n+2^j k} \tag{26}$$

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.

2.4 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 (18-24). 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 :

$$V_{j+1} \otimes V_{j+1} = (V_j \oplus W_j) \otimes (V_j \oplus W_j) = (W_j \otimes W_j) \oplus (W_j \otimes V_j) \oplus (V_j \otimes W_j) \oplus (V_j \otimes V_j) \tag{27}$$

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 (27) :

$$W_j(\mathbb{T}^2) = (W_j \otimes W_j) \oplus (W_j \otimes V_j) \oplus (V_j \otimes W_j) \tag{28}$$

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

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 $V_{j-1}(\mathbb{T}^2)$:

$$\alpha_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} g_k^j g_m^j s_{k+2i,m+2l}^j \tag{29}$$

$$\beta_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} g_k^j h_m^j s_{k+2i,m+2l}^j \tag{30}$$

$$\gamma_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} h_k^j g_m^j s_{k+2i,m+2l}^j \tag{31}$$

$$s_{i,l}^{j-1} = \sum_{k,m=0}^{2^j-1} h_k^j h_m^j s_{k+2i,m+2l}^j \tag{32}$$

où (g_k^j) et (h_k^j) sont les filtres unidimensionnels introduits en 2.3 (25-26). 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 $V_{j-1}(\mathbb{T}^2)$) :

$$s_{s,l}^j = \sum_{n,m=0}^{L-1} \left(h_{s-2n}^j h_{l-2m}^j s_{n,m}^{j-1} + g_{s-2n}^j g_{l-2m}^j \alpha_{n,m}^{j-1} + g_{s-2n}^j h_{l-2m}^j \beta_{n,m}^{j-1} + h_{s-2n}^j g_{l-2m}^j \gamma_{n,m}^{j-1} \right) \tag{33}$$

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.

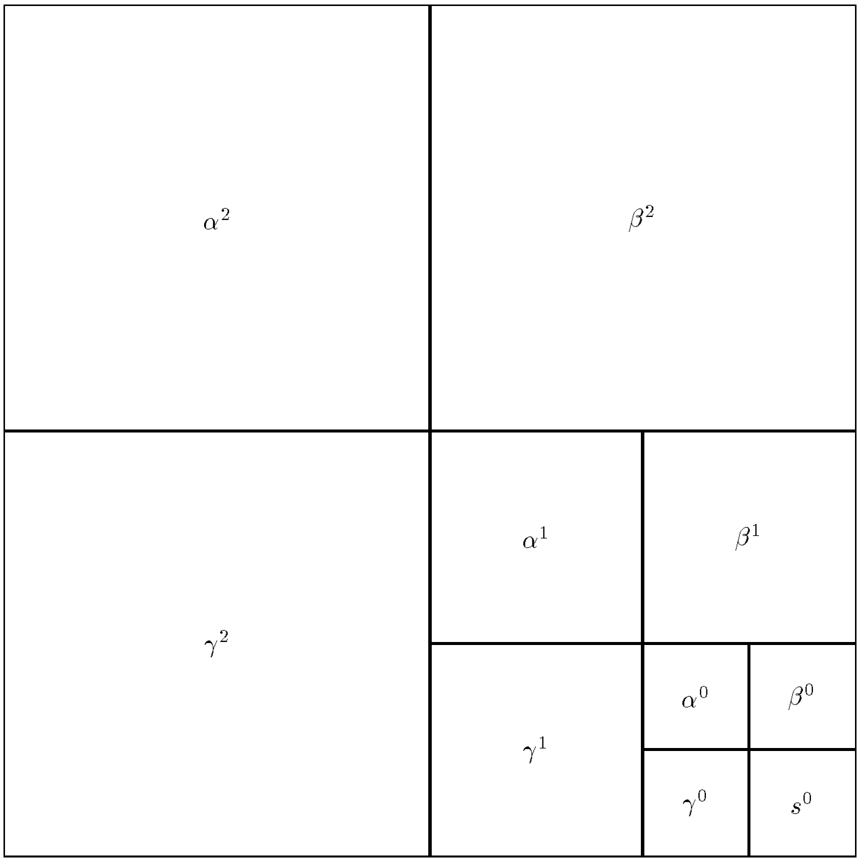

FIG. 1 - *Analyse en ondelettes de $L^2(\mathbb{T}^2)$ d'un champ bidimensionnel : la taille des carrés est proportionnelle au nombre de coefficients contenus dans ces carrés.*

2.5 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 (2.3), 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.

$$Q_p = Q_p^1 Q_p^2 Q_p^3 \cdots Q_p^p$$

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

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. A défaut d'être mesurée sur un exemple pratique, 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. Si donc on envisage la TOD d'une fonction l'interprétation est directe, elle nous est fournie par les théorèmes liant la décroissance des coefficients d'ondelettes à la régularité d'une fonction [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.

3 Problèmes modèles

3.1 Problèmes continus

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

Trouver $u(x,t)$ tel que

$$\begin{cases} \partial_t u + L_x(u) = 0 \quad \text{pour } t > 0 \\ u(x,0) = u_0(x) \end{cases} \tag{34}$$

où L_x est un opérateur différentiel linéaire en x. Dans les exemples numériques présentés, L_x sera l'opérateur de diffusion $-\nu\partial_x^2$, ou l'opérateur de convection $v(x)\partial_x$.

Comme indiqué plus haut on considère un schéma de discrétisation en temps implicite, par exemple le schéma d'Euler rétrograde :

$$\frac{u^{n+1} - u^n}{\delta t} + L_x(u^{n+1}) = 0$$

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 :

$$\mathcal{A}u^{n+1} = u^n \quad \text{avec} \quad \mathcal{A} = (1 + \delta t L_x) \tag{35}$$

Notre problème modèle continu sera donc : Étant donnée $f \in L^2(\mathbb{T})$, trouver $u \in \mathcal{H}(\mathbb{T})$ tel que :

$$\mathcal{A}u = f \;\; \text{avec} \;\; \mathcal{A} = (1 + \tau L_x). \tag{36}$$

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

$$u = \mathcal{B}f \quad \text{avec} \quad \mathcal{B}f = \int_{\mathbb{T}} G(x,y) f(y) dy \tag{37}$$

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

3.2 Problèmes discrets

Une discrétisation en espace ramène le problème continu (36) 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 :

$$AU = F \tag{38}$$

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 sur ce sous-espace et la matrice A de dimension $N \times N$ est définie par : $a_{ij} = < \mathcal{A}\varphi_j | \; \varphi_i >$.

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

$$U = BF \tag{39}$$

La matrice B étant cette fois définie par:

- pour une méthode de Galerkin:

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

- pour une méthode de différences finies:

  b_{ij} est une approximation de $G(x_i, x_j)\delta x_i \delta x_j$.

D'autres approches, liées aux méthodes de Petrov-Galerkin et aux bases biorthogonales d'ondelettes permettent la discrétisation de (36) et sont développées dans [Liandrat et al.92, Dahlke et al.94].

3.3 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 (38) 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).

Dans cet article, nous nous intéresserons à la formulation (39), 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 chapitre B) 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 [Meyer90, Beylkin et al.91]. Notre but est donc, en utilisant cette propriété, de transformer B pour la rendre creuse. Diverses méthodes 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.

4 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 ce chapitre est d'étudier différentes décompositions permettant de rendre la matrice B et (ou) les vecteurs X, Y creux, de façon à optimiser le coût du produit matrice-vecteur BX. Les méthodes présentées dans ce chapitre ont déjà fait l'objet d'études théoriques dans [Beylkin et al.91, Dahmen et al.94].

4.1 Analyse sur la base de Fourier

Cette analyse (classique) s'applique seulement quand la matrice B est circulante ; dans ce cas le produit $Y = BX$ s'écrit simplement comme un produit de convolution :

$$y_n = \sum_{i=0}^{N-1} b_{n-i} \cdot x_i \tag{40}$$

Si l'on a posé :

$$B = (b_{i-j})_{i,j=0,N-1} \ , \ Y = (y_i)_{i=0,N-1} \text{ et } X = (x_i)_{i=0,N-1} \tag{41}$$

les b_i étant périodiques de période N.

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

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

$\hat{X} = (\hat{x}_k)_{k=0,N-1}$ 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 :

$$\hat{B} = \left(diag(\hat{b}_k)\right)_{k=0,N-1} \tag{42}$$

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.

4.2 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 (voir notations du chapitre 2.5) :

- $Y' = Q_pY$ la TOD de Y
- $X' = Q_pX$ la TOD de X
- $B' = Q_pBQ_p^{-1}$

Les vecteurs Y' et X' seront creux si les fonctions associées sont régulières au sens des ondelettes (voir section 2.5).

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 2.5), 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

$$B_{k,l}^{i,j} = \int_T \psi_0^i(x - 2^{-i}k)\mathcal{B}\psi_0^j(x - 2^{-j}l)dx \ , \ 0 \leq k \leq 2^i - 1 \ , \ 0 \leq l \leq 2^j - 1 \tag{43}$$

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) (figures 3 et 4). Pour les exemples numériques tirés de la mécanique des fluides (section 6), les vecteurs X et Y seront très bien représentés en ondelettes, dans le sens où peu de coefficients d'ondelettes seront non négligeables. Ainsi le produit $Y = BX$ nécessitera peu d'opérations.

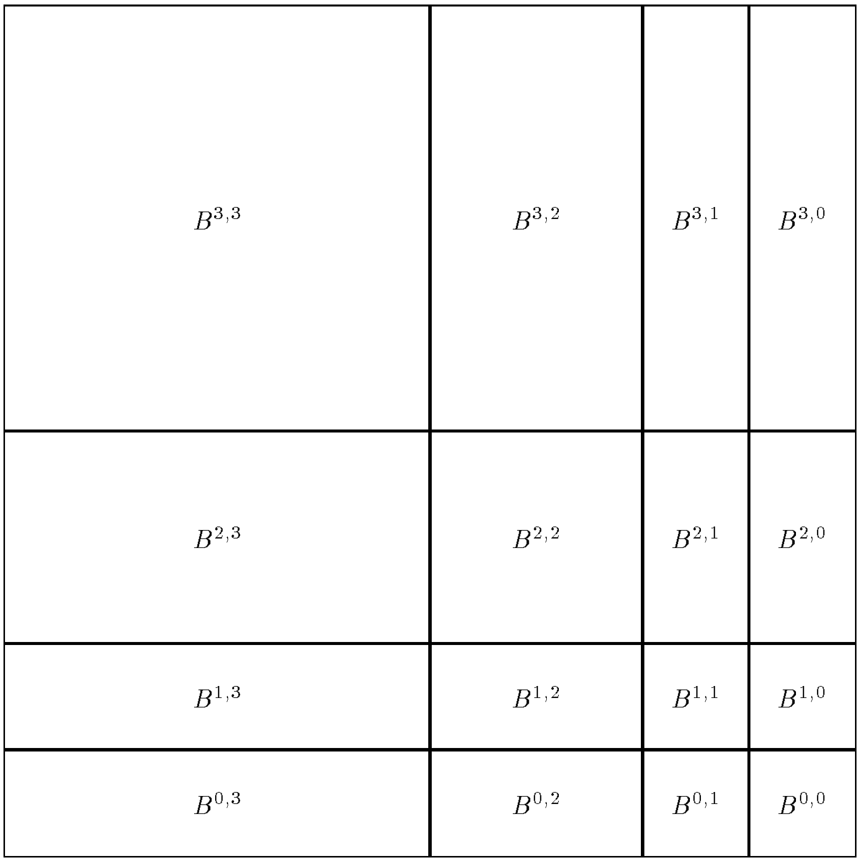

FIG. 2 - *Structure de la matrice B' en base d'ondelettes, chaque bloc $B^{i,j}$ correspond à* $B^{i,j} = \Pi_{W_i} B \Pi_{W_j}$

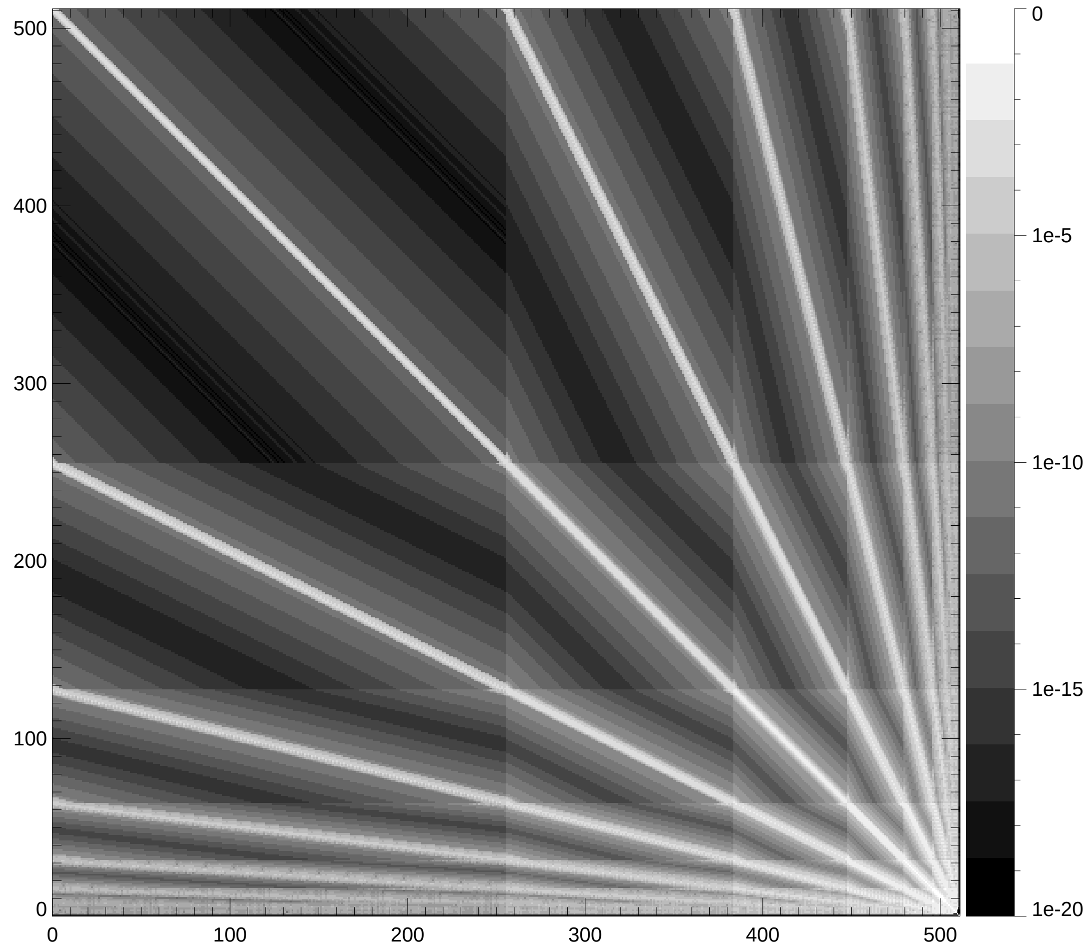

FIG. 3 - *Représentation de la matrice B en ondelettes (à support compact d'ordre 4) où B est la matrice de l'opérateur: $\mathcal{B} = (1 - \nu\delta t\partial_{xx}^2)^{-1}$ (échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

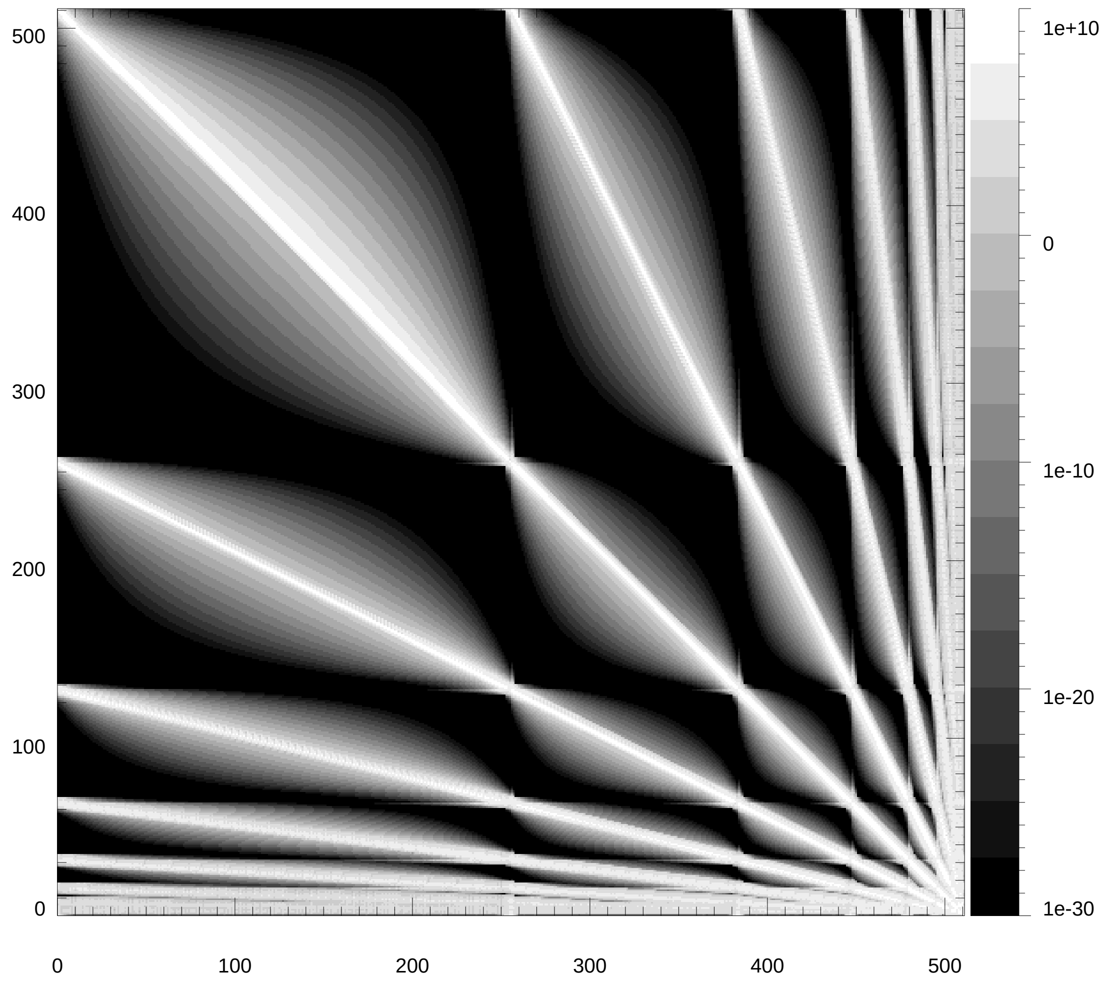

FIG. 4 - *Représentation de la matrice B en ondelettes (à support compact d'ordre 4) où B est la matrice de l'opérateur :* $\mathcal{B} = (1 + \sin \pi x \partial_x)^{-1}$ *(échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

4.3 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 (figures 6, 7).

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 2.5) dans V_p aux vecteurs Y et X et à la matrice B, de telle sorte que

$$u_p = \mathcal{B}_p f_p \tag{44}$$

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 (44) 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 :

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

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 :

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

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 2.5) 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 :

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

L'algorithme BCR est donc le suivant :

Étape 0: La matrice B est transformée par la TOD bidimensionnelle décrite en section 2.4. 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 $0 \leq j \leq p-1$ et $\Pi_{V_0} B \Pi_{V_0}$ (voir figure 2).

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 5.

Étape 3: Le vecteur ainsi obtenu est « recompacté » en utilisant une étape de l'algorithme de synthèse en ondelettes, représenté par l'équation (17), à savoir :

$$\tilde{y}_{1,k} = \sum_n g_{k-2n} y_{0,n} + \sum_n h_{k-2n} z_{0,n} \tag{45}$$

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

$$\begin{array}{ccc} Y_0 & \longrightarrow & \tilde{Y}_1 \\ Z_0 & \nearrow & \end{array} \tag{46}$$

On additionne ensuite Y_1 et $\tilde{Y}_1$ puis on itère l'étape (46) :

$$\begin{array}{ccc} Y_1 + \tilde{Y}_1 & \longrightarrow & \tilde{Y}_2 \\ Z_1 & \nearrow & \end{array} \tag{47}$$

Jusqu'à obtenir $\tilde{Y}_p = Y$.

<u>Remarque</u> : 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 5 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.

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.

5 Analyse en ondelettes et BCR 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 L_x du problème (34) 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 en dimensions supérieures. Nous exposons en premier la méthode BCR car celle-ci est plus simple.

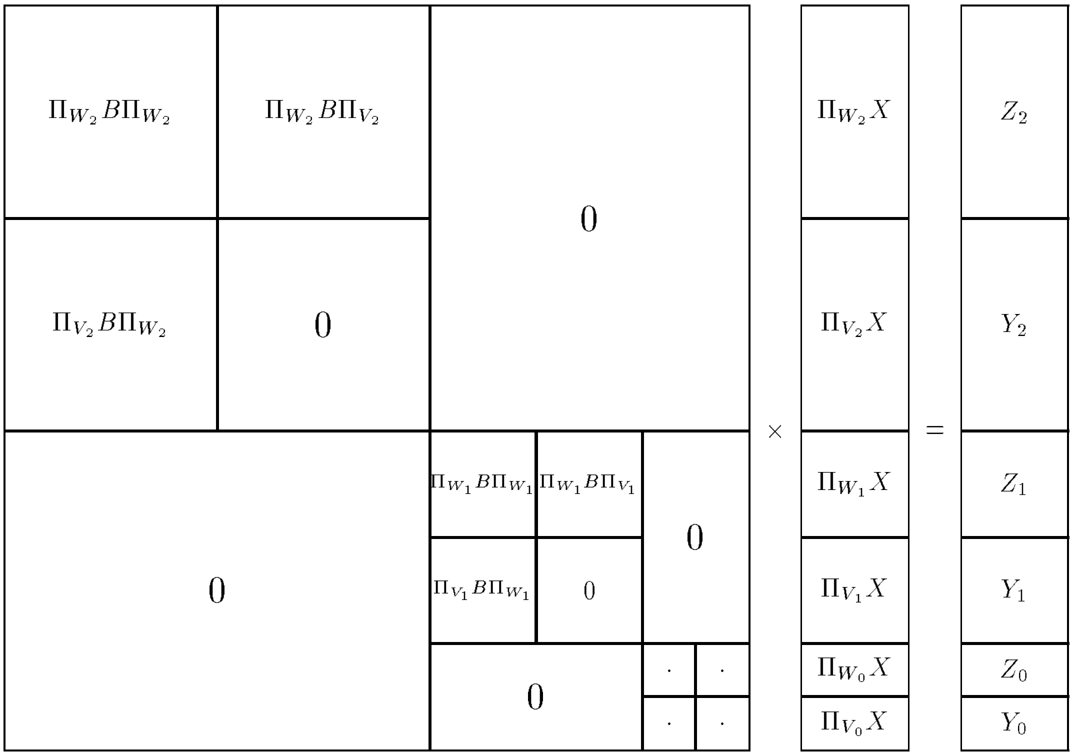

FIG. 5 - *Structure de la matrice B et du vecteur X décomposés par l'algorithme BCR*

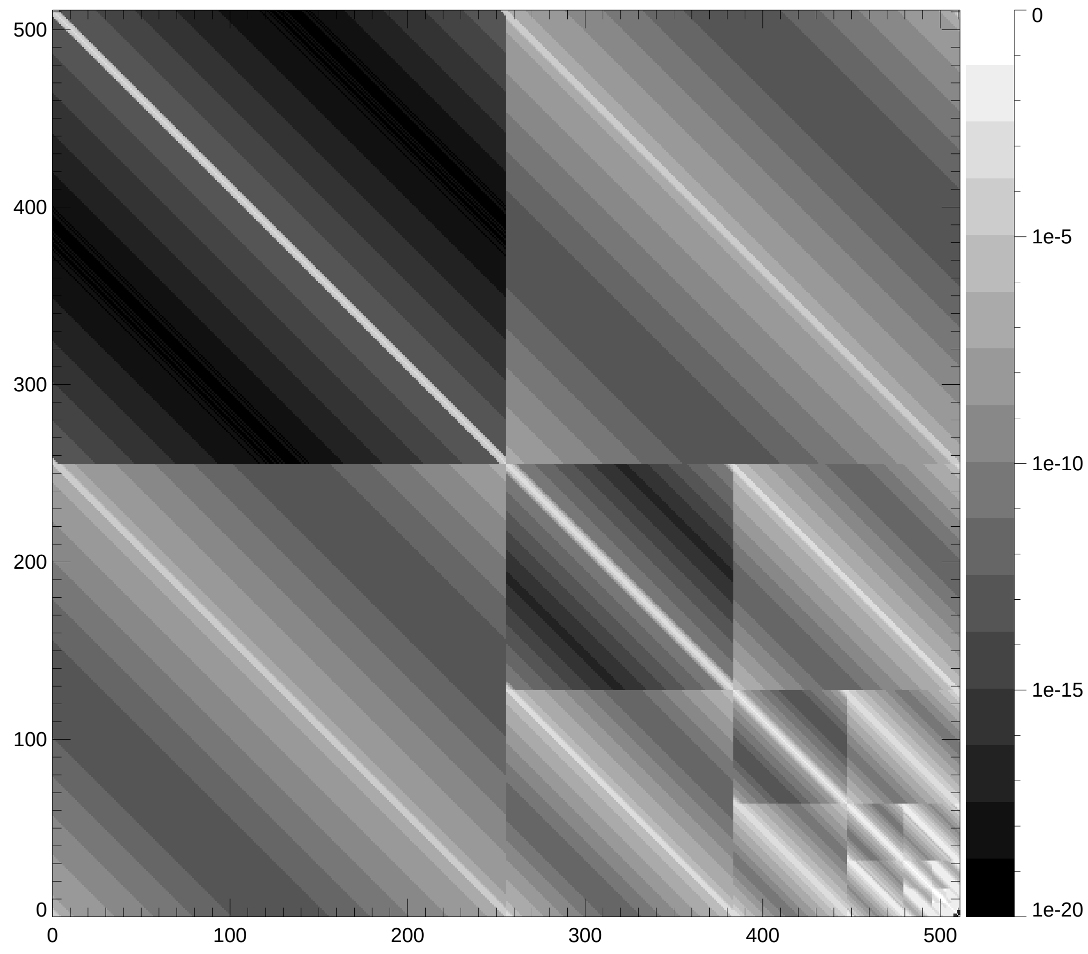

FIG. 6 - *Représentation BCR de la matrice B où B est la matrice de l'opérateur:* $\mathcal{B} = (1 - \nu\delta t\partial_{xx}^2)^{-1}$ *(échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

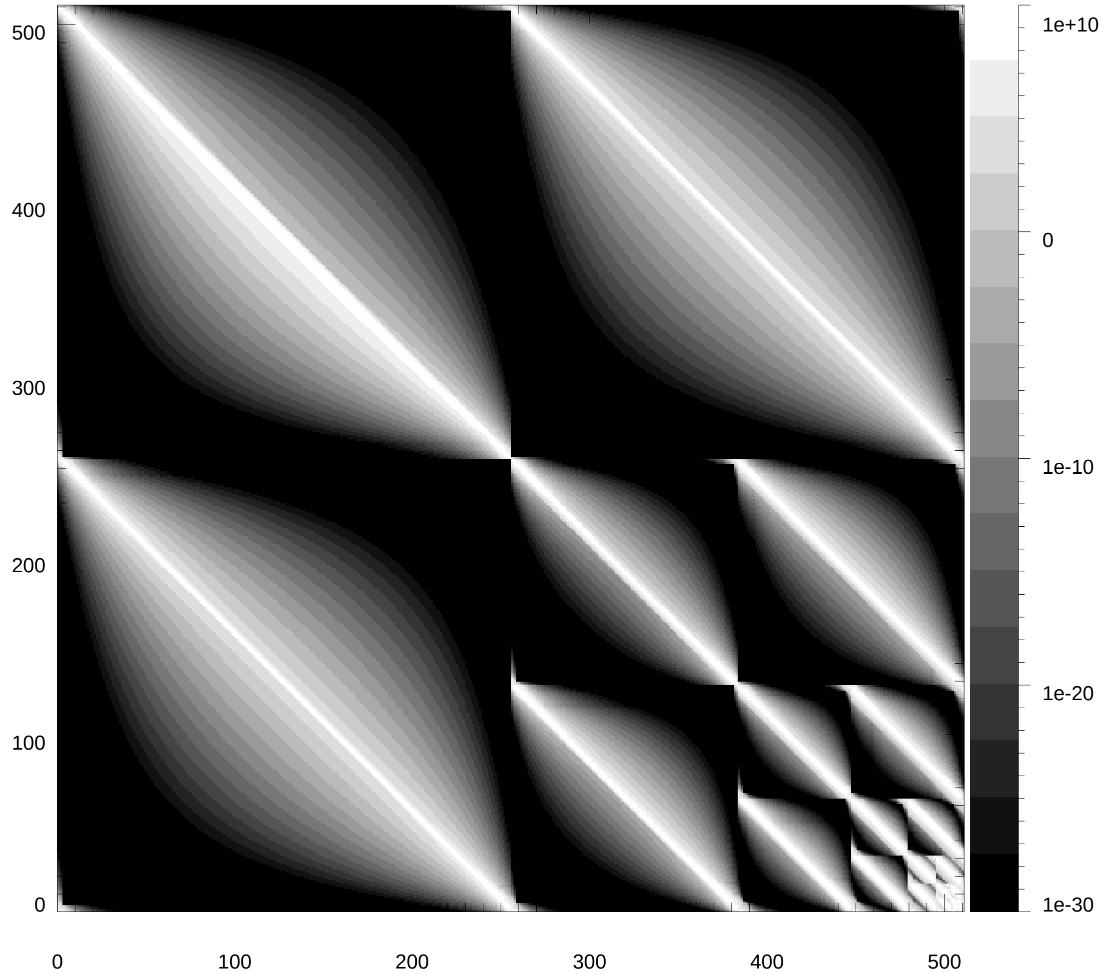

FIG. 7 - *Représentation BCR de la matrice B où B est la matrice de l'opérateur :* $\mathcal{B} = (1 + \sin \pi x \partial_x)^{-1}$ *(échelle de couleurs logarithmique, les coefficients significatifs sont de couleurs claires)*

5.1 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 (29–32):

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

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 :

$$s^{j-1}(n) = s^{j-1}_{0,n} = s^j(2n) + \sum_{k=1}^{2^j-1} H^j_k \cdot \left(s^j(2n+k) + s^j(2n-k)\right) \text{ avec } H^j_k = \sum_{l=0}^{2^j-k} h^j_l h^j_{l+k} \tag{48}$$

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 (10-11), 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 (48).
On a également (voir [Daubechies92]) :

$$H^j_{2k} = 0, \text{ pour } j = 0, \cdots, p-1 \text{ et } k = 1, \cdots, 2^{j-1} \tag{49}$$

Remarque :

- Avec le choix de g (12), 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 (49) 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.

5.2 Décomposition en ondelettes

La matrice B, transformée par la méthode standard décrite à la section 4.2, est composée de blocs qui ne sont pas carrés (figure 2). 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) d'ordre $2^j \times 2^i$ vérifient : si $i \geq j$

$$B^{i,j}_{k,l} = \int_0^1 \psi^i_0(x - \frac{k}{2^i}) \;\; \mathcal{B}\psi^j_0(x - \frac{l}{2^j}) \; dx = \int_0^1 \psi^i_0(y - \frac{k - 2^{i-j}l}{2^i}) \;\; \mathcal{B}\psi^j_0(y) \; dy = f^{i,j}_{k-2^{i-j}l} \tag{50}$$

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 quatres termes se calculent de manière analogue, détaillons le calcul de $sd^{j-2,i}$.

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

   On a donc :

$$sd^{j-2,i}(m) = \sum_{l=0}^{2^{j-1}-1} g_l \;\; sd^{j-1,i}(m - 2^{i-j+1}l) \tag{51}$$

On réitère ensuite l'étape 1 pour décomposer $ss^{j-1,j-1}$. Ces deux étapes sont schématisées à la figure 8.
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 sera détaillé au chapitre 6.

6 Calcul pratique du produit matrice-vecteur

Dans la pratique, le produit $Y = BX$ est calculé 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 $\epsilon_X \|X\|$. Nous avons vu dans les exemples illustrés aux figures (3–7) que pour les deux décompositions (ondelettes et méthode BCR), la matrice B a beaucoup de coefficients voisins de 0. La mise en œuvre du produit BX tient compte de cette « lacunarité », ainsi que d'une éventuelle lacunarité du vecteur X :

– La partie 6.1 est consacrée au cas général, pour une matrice non circulante.

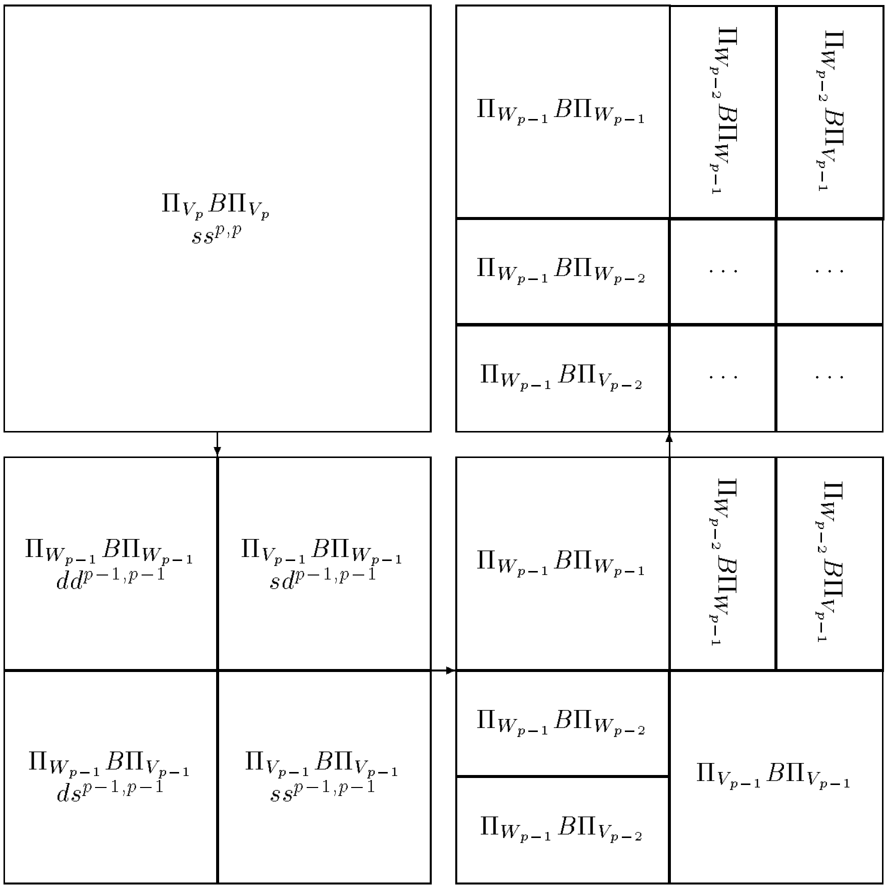

FIG. 8 - *Schéma de décomposition en ondelettes d'une matrice circulante*

– La partie 6.2 détaille le produit BX dans le cas où B est circulante, stockée en tenant compte des deux propriétés : B circulante et B creuse.

Nous estimons dans les deux cas le coût théorique du produit matrice-vecteur, puis l'erreur finale sur le vecteur $Y = BX$ en fonction de ϵ_B et ϵ_X (partie 6.3).

6.1 Cas général

6.1.1 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 élément 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.

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 V, la matrice étant parcourue verticalement; les numéros des lignes correspondantes sont stockés dans un deuxième tableau C, enfin un troisième tableau D sert à repérer le début de chaque colonne; le premier élément de la colonne i est $V(D(i))$. 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 :

$$\begin{array}{lccccccc} V = & 2. & 12. & 1. & 3. & 5. & 4. & \\ C = & 1 & 3 & 4 & 3 & 2 & 1 & 0 \\ D = & 1 & 4 & 5 & 6 & 7 & & \end{array}$$

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

6.1.2 Produit matrice-vecteur

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

```
y=0
i=1
   tant que cx(i) <> 0
     Pour j:=d(cx(i)) jusqu'à d(cx(i)+1)-1
          Y(c(j)):=Y(c(j))+V(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.

6.2 Cas d'une matrice circulante

6.2.1 Stockage

Dans le cas d'une matrice B circulante, nous avons vu en section 5 que sa décomposition en ondelettes comportait également des propriétés de circularité : on peut exploiter celles-ci pour diminuer le volume du stockage.

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 *colv* et *valv*,
colv contenant le numéro de ligne d'un élément non nul,
valv contenant la valeur de cet élément.

Décomposition en ondelettes : On a besoin de stocker seulement un vecteur par bloc de la matrice représentée en figure 2. 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 :

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

Est stockée sous la forme :

$ind =$	1		3		5		7	8		10	11	12	13		15	16	17	18	19	20	21
$col =$	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.

Décomposition type BCR : 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, mais cette fois les blocs sont tous carrés.

6.2.2 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:=colv(cx)
       x:=valv(cx)
       Pour i:=0,j-1
          Pour cy:=ind(j*p+i),ind(j*p+i+1)-1
```

```
            c:=col(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:=col(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.

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

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

On a noté :
$$\|b_i\| = (\sum_{j=1,N} b_{i,j}^2)^{\frac{1}{2}} \text{ et } \|B\| = (\sum_{i=1,N} (\sum_{j=1,N} b_{i,j})^2)^{\frac{1}{2}}. \tag{52}$$

Une majoration de l'erreur relative est donc :
$$\frac{\|Y - \bar{Y}\|^2}{\|X\|^2} = O\left(N^2 \epsilon_B^2 + N\epsilon_X^2 \|B\|^2\right) \tag{53}$$

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.94] pour des seuils ϵ_B différents suivant les blocs présents dans la matrice (voir la structure de la matrice B figures 2 et 5).

7 Expériences numériques

Nous présentons dans cette partie une comparaison des différentes méthodes pour la résolution numérique du problème (34) développé au chapitre 3. Les temps de calculs présentés dans ce chapitre correspondent à des tests numériques effectués sur HP 720.

1. La somme porte sur les termes tels que $|b_{i,j}| < \epsilon_B$ ou $|x_j| < \epsilon_X \|X\|$

7.1 Présentation des problèmes test

7.1.1 Équation de diffusion

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

$$\left\{ \begin{array}{l} \partial_t u - \nu\ \partial_{xx} u = 0 \quad x \in [0,1]\ , t \geq 0 \\ u(x,0) = u_0^1(x) = \left\{ \begin{array}{ll} x & \text{si } x \in [0, \frac{1}{2}[\\ x-1 & \text{si } x \in [\frac{1}{2}, 1[\end{array} \right. \end{array} \right. \tag{54}$$

Après discrétisation en temps (schéma d'Euler implicite) et en espace (différences finies d'ordre deux, ce qui est équivalent à des ondelettes à support compact d'ordre 1), (54) s'écrit :

$$U^{n+1} = B_1 U^n \tag{55}$$

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 9.

7.1.2 Équation de transport

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

$$\left\{ \begin{array}{l} \partial_t u + \sin(\pi x)\ \partial_x u = 0 \quad , x \in [0,1]\ ,\ t \geq 0 \\ u(x,0) = u_0^2(x) \end{array} \right. \tag{56}$$

$u_0^2(x)$ 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 :

$$U^{n+1} = B_2 U^n \tag{57}$$

B_2 est représenté à la figure 10.

7.2 Ajustement des paramètres

- Les systèmes (55) et (57) 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 discrétisations (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 (54) et (56).

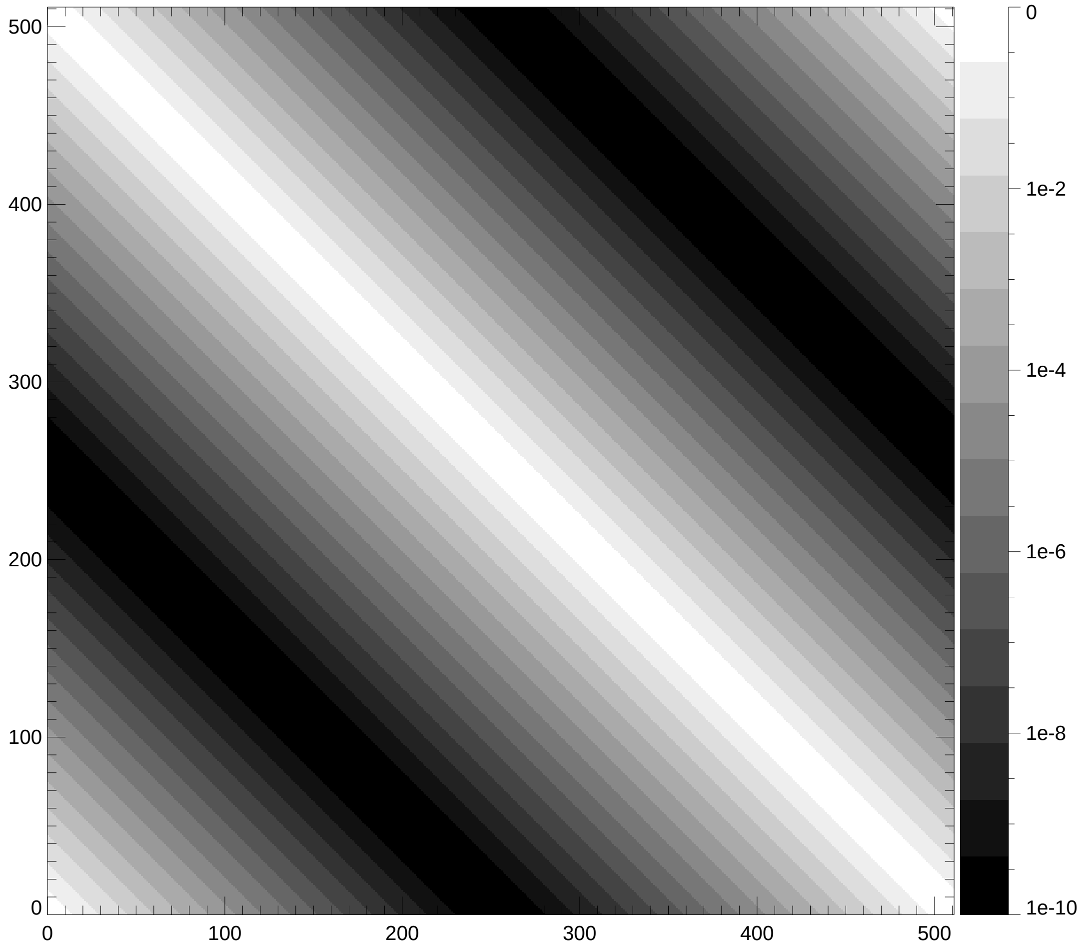

FIG. 9 - *Représentation matricielle de l'opérateur* $(I-\Delta)^{-1}$ *(échelle logarithmique)*

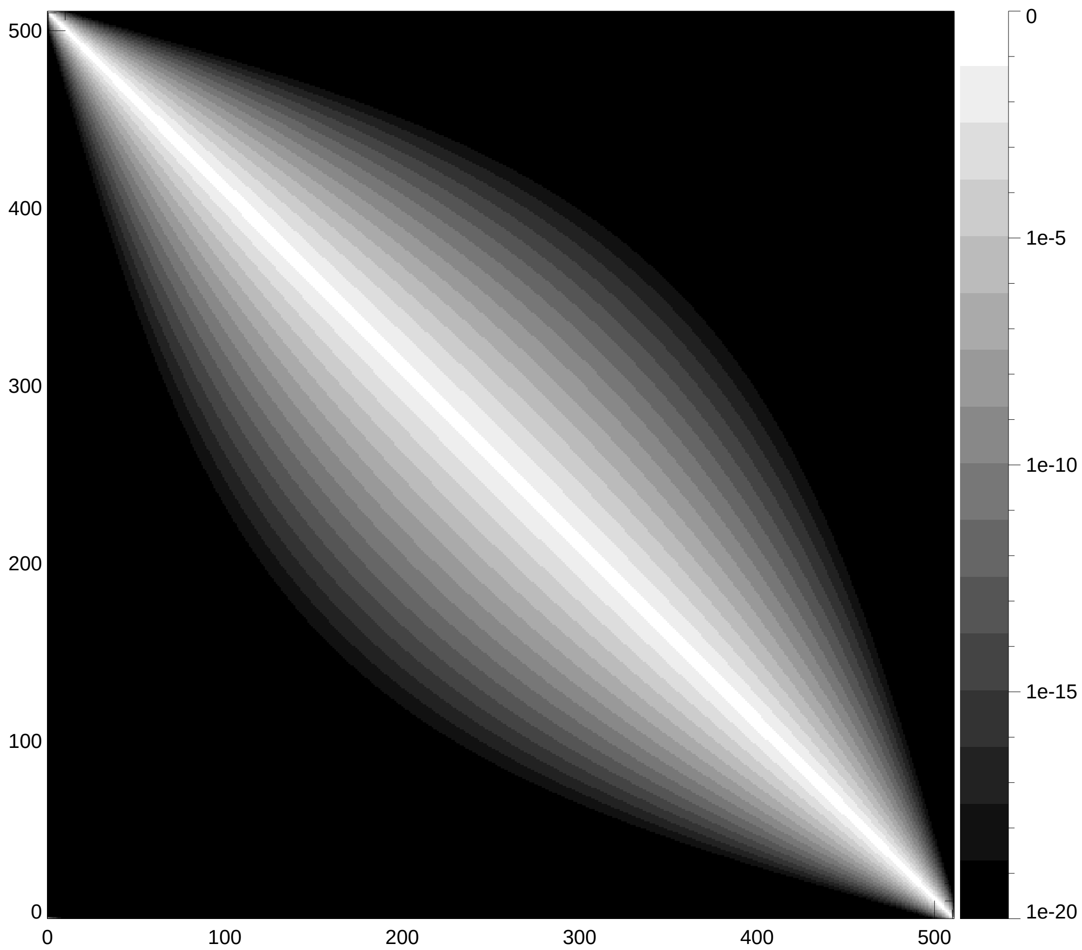

FIG. 10 - *Représentation matricielle de l'opérateur* $(I - \sin(\pi x)\partial_x)^{-1}$ *(échelle logarithmique)*

7.3 Comparaison des lacunarités

7.3.1 En fonction de l'ordre de la matrice

Les figures (11–12) 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 l'opérateur. 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).

7.3.2 En fonction de l'ordre de l'ondelette

Les figures 13 et 14 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 $N = 1024$, 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.

7.3.3 En fonction du seuil sur la matrice

Les figures (15–16) 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 13 et 14).

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

7.4 Calculs d'erreurs en fonctions des seuils ϵ_B sur l'opérateur et ϵ_U sur la solution — Temps de calcul

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 17-18).

7.4.1 Équation de la chaleur par la méthode standard

Dans le tableau 19, 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 (54) pour $t = 0.1$. En avant-dernière colonne figure l'erreur (en norme l^2) avec la solution de référence

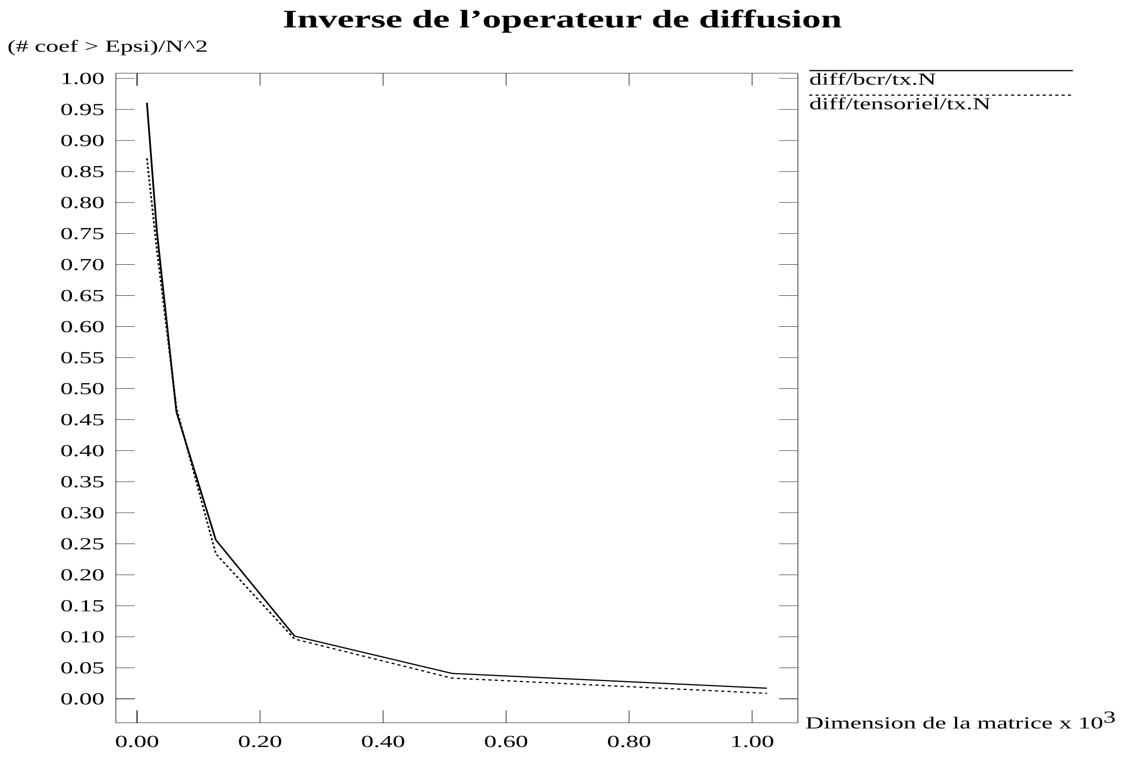

FIG. 11 - *Taux de compression de l'opérateur B_1 en fonction de l'ordre de la matrice (seuil $\epsilon_B = 10^{-5}$, ondelettes d'ordre 4).*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

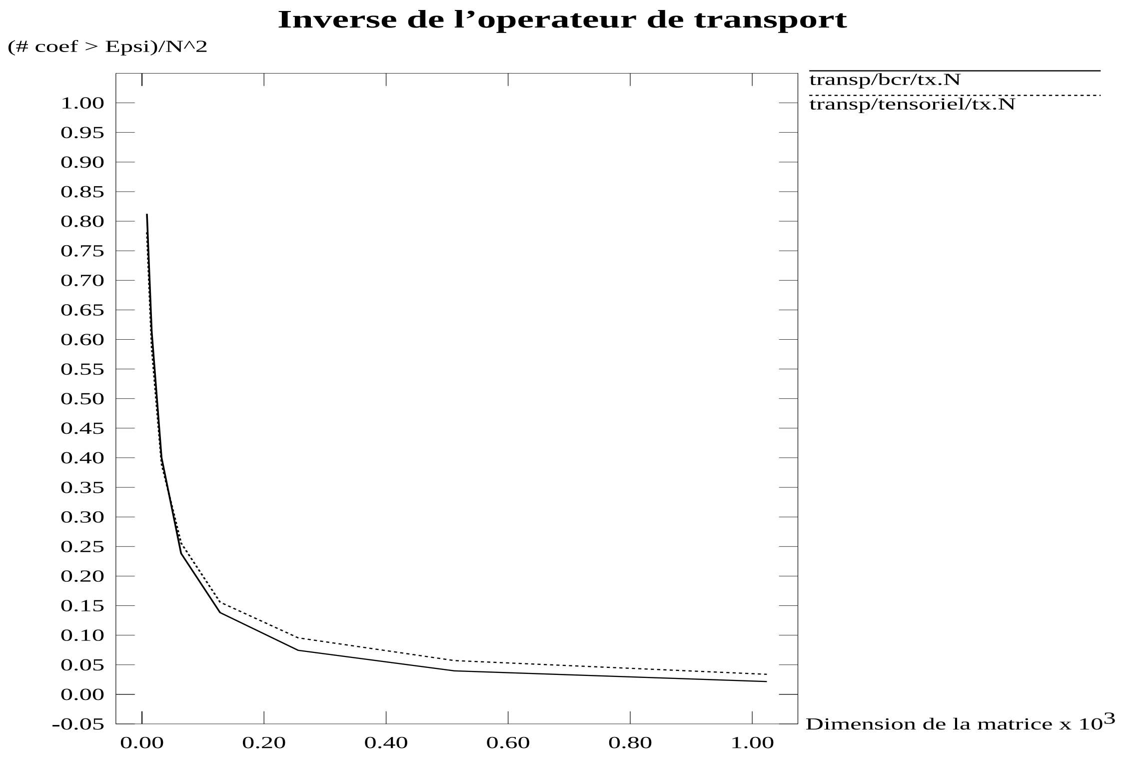

FIG. 12 - *Taux de compression de l'opérateur B_2 en fonction de l'ordre de la matrice (seuil $\epsilon_B = 10^{-5}$, ondelettes d'ordre 4).*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

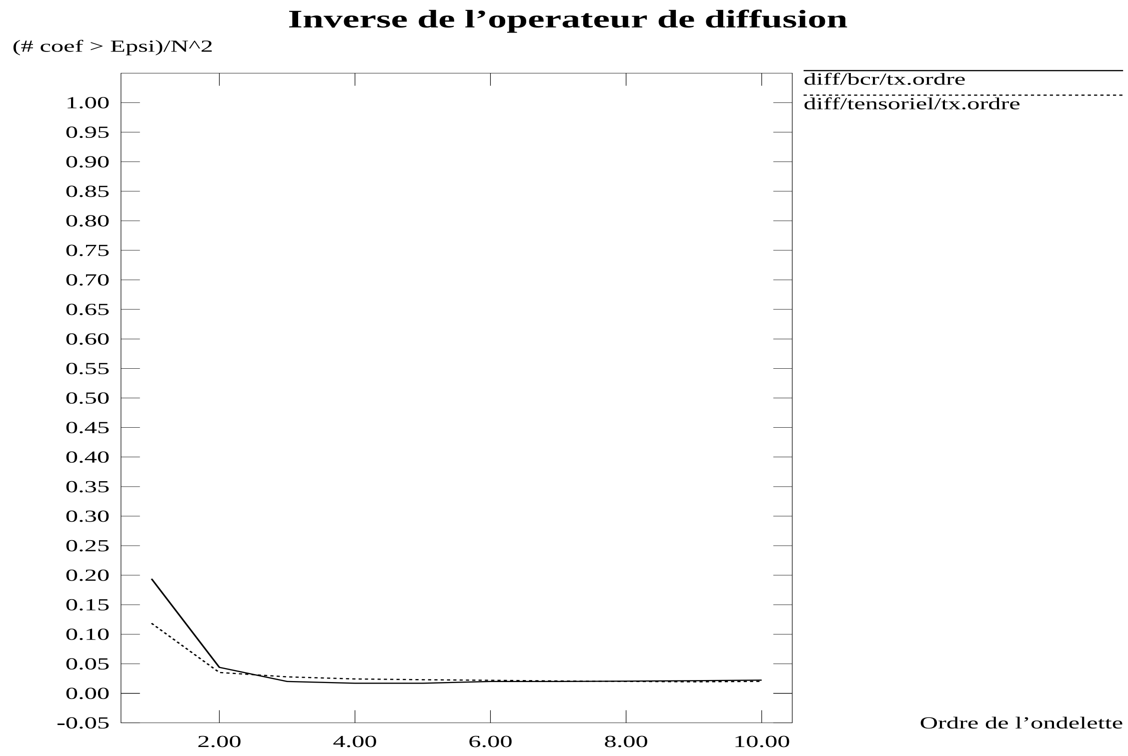

FIG. 13 - *Taux de compression de l'opérateur* B_1 *en fonction de l'ordre de l'ondelette de base (seuil* $\epsilon_B = 10^{-5}$, *N=1024)*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

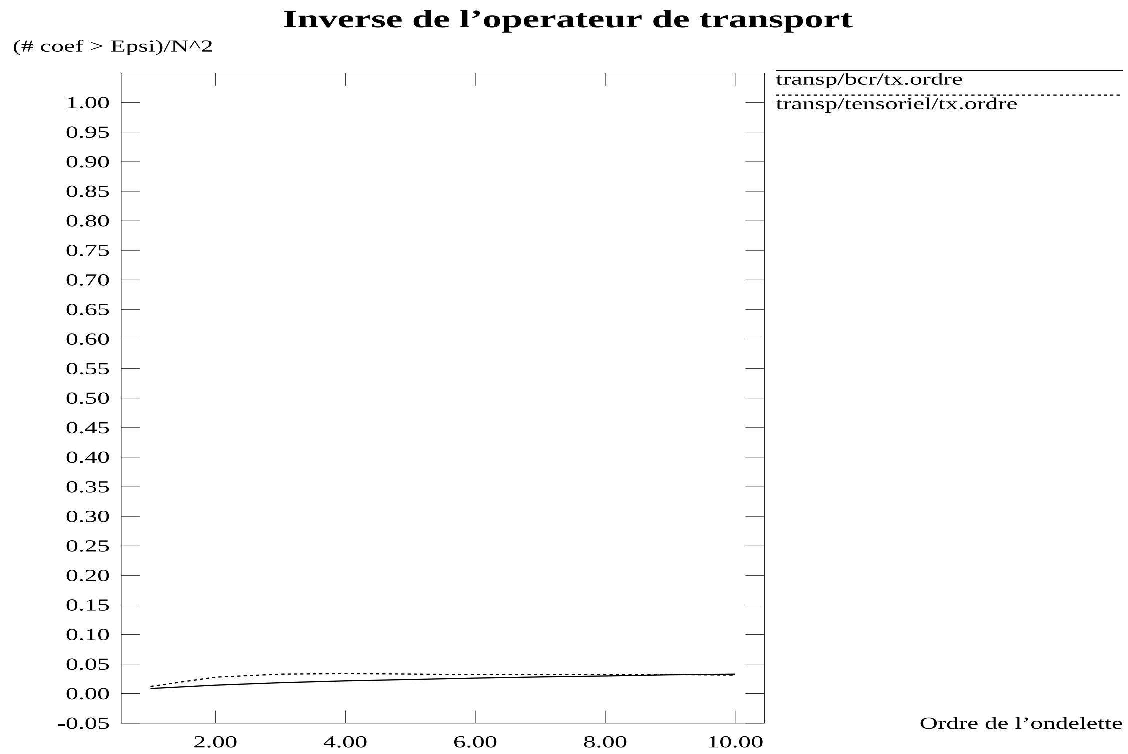

FIG. 14 - *Taux de compression de l'opérateur* B_2 *en fonction de l'ordre de l'ondelette de base (seuil* $\epsilon_B = 10^{-5}$, *N=1024)*
1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard

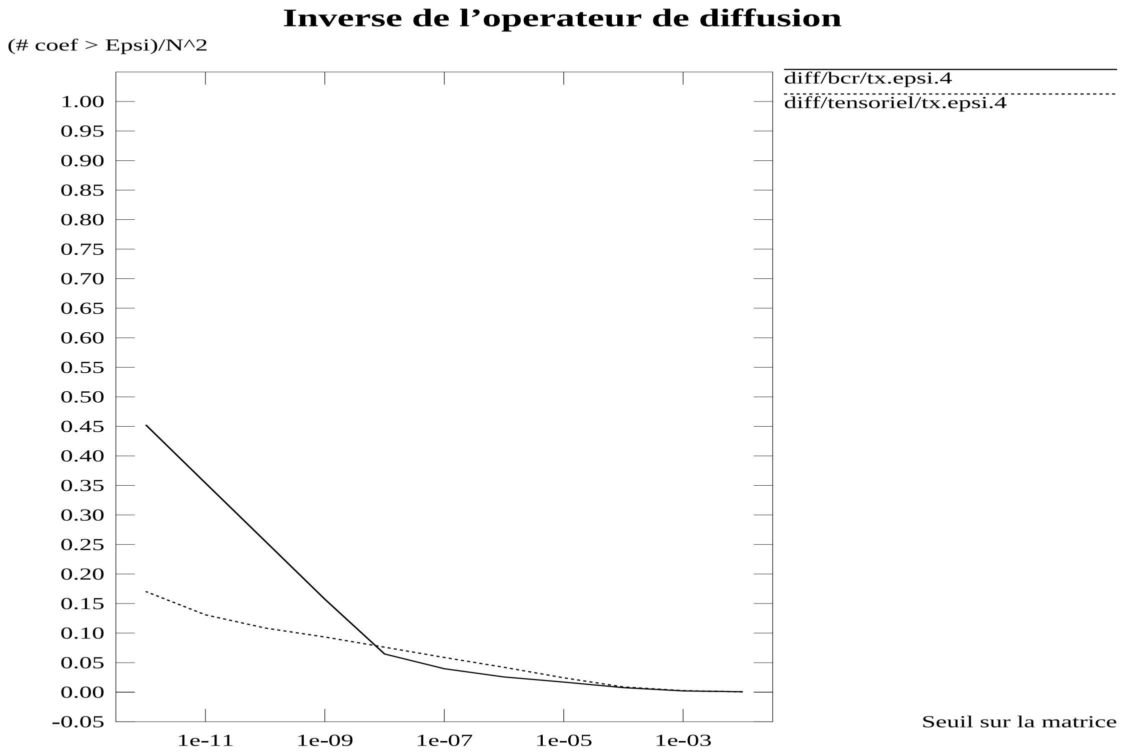

FIG. 15 - *Taux de compression de l'opérateur B_1 en fonction du seuil sur la matrice ϵ_B (N=1024) 1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard*

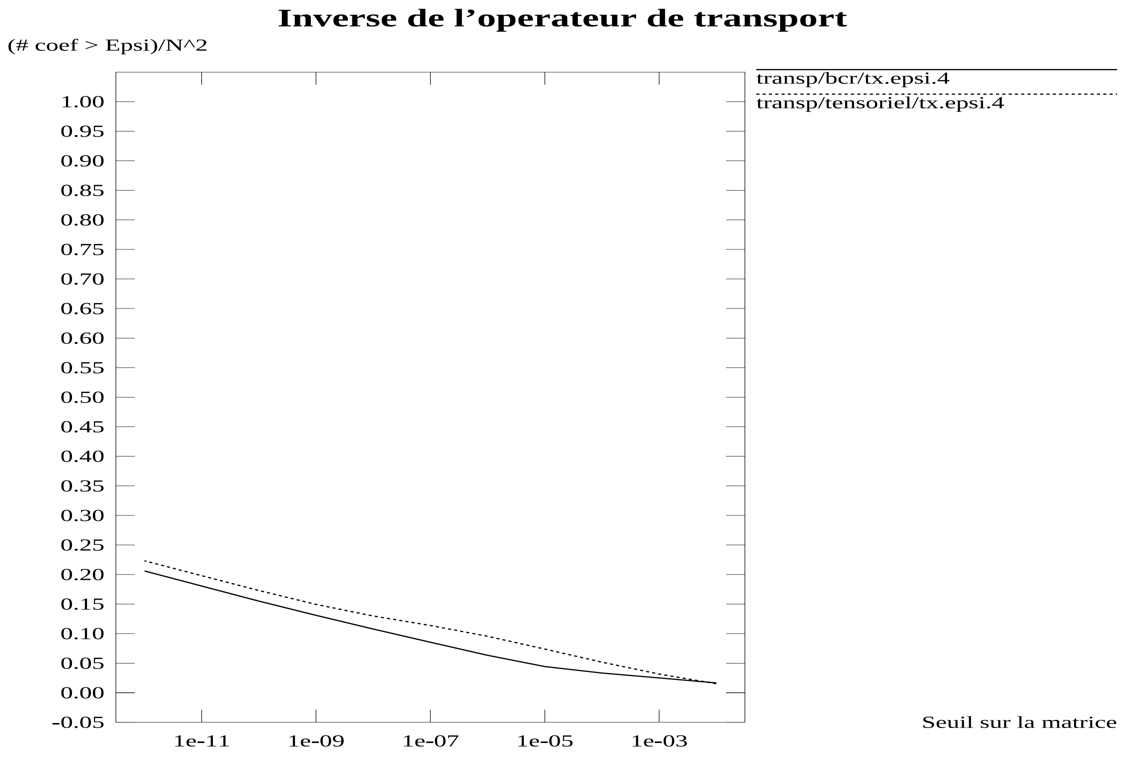

FIG. 16 - *Taux de compression de l'opérateur B_2 en fonction du seuil sur la matrice ϵ_B (N=1024) 1) Trait plein : forme non standard (BCR) 2) Pointillé : forme standard*

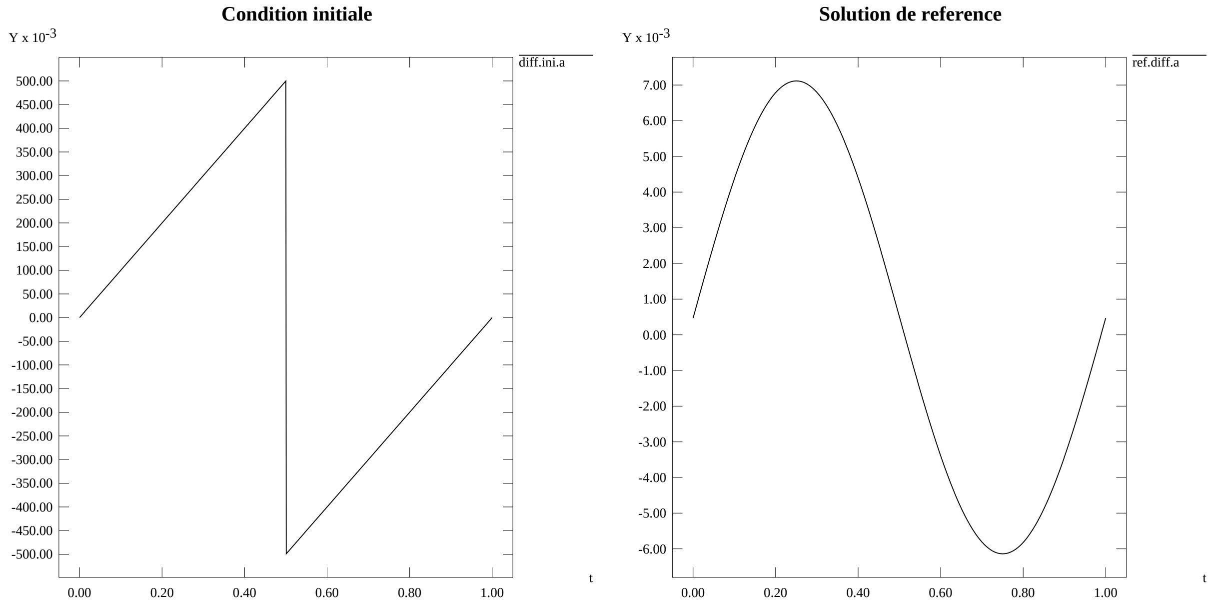

FIG. 17 - *Condition initiale et solution de référence au temps $t = 0,1$ (Équation de diffusion)*

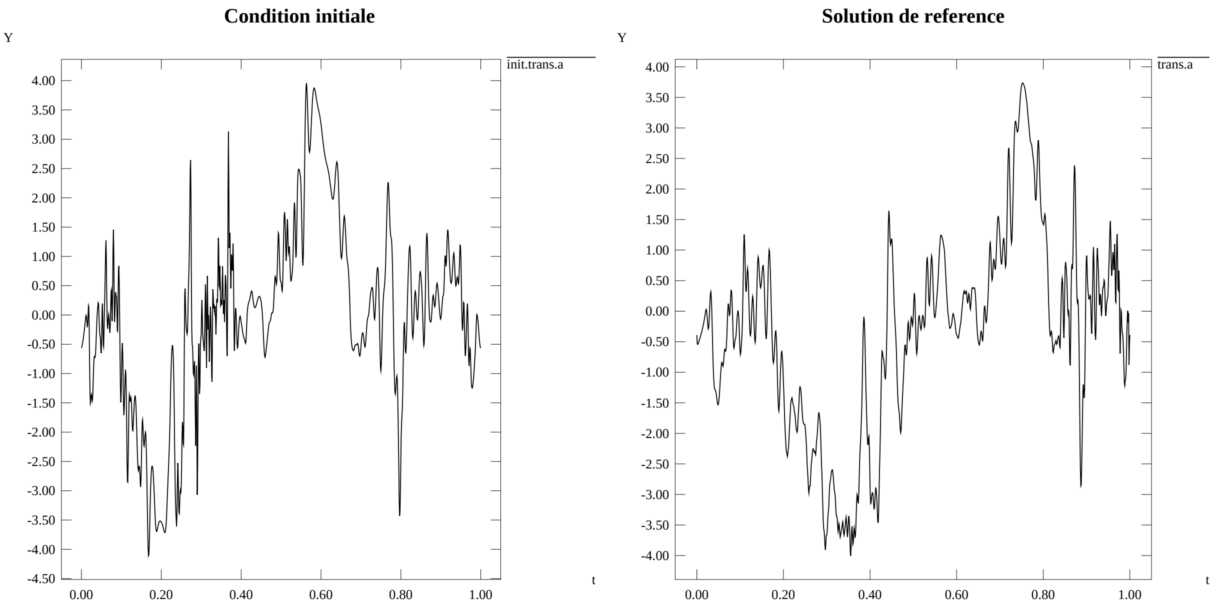

FIG. 18 - *Condition initiale et solution de référence au temps $t = 0,2$ (Équation de transport)*

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.

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. 19 - *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 (54) 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 $\epsilon_B = 10^{-8}$).

7.4.2 Équation de la chaleur par la méthode BCR

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, 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 19) 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. 20 - *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)*

7.4.3 Équation de transport par la méthode standard

Ce tableau est analogue au tableau 19, l'opérateur B_1 étant remplacé par l'opérateur B_2 décomposé en ondelettes. Sur ce tableau 21, 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.

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}	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. 21 - *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)*

7.4.4 Équation de transport par la méthode BCR

Ce tableau est analogue au tableau 19, 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 20 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 veceur U dansl'espace physique.

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}	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. 22 - *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)*

7.5 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 (55) et (57) 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 (55) et (57) (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 $r = 10$.

7.5.1 Équation de la chaleur – Méthode standard

Pour cette équation, d'après la figure 19, 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. 23 - *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 25).

7.5.2 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. 24 - *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 25).

8 Conclusion et perspectives

Dans cet article 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

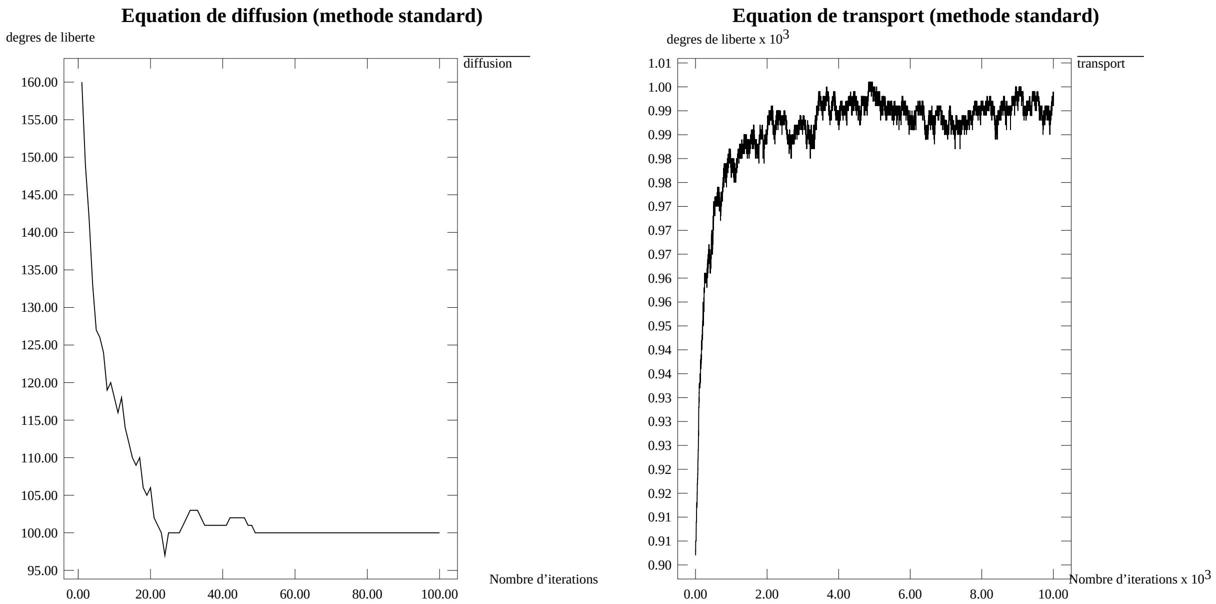

FIG. 25 - *Évolution du nombre de degrés de liberté (coefficients de la solution $> 10^{-3}$) en fonction du nombre d'itérations*

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. Ainsi n'est-elle pas adaptée au problème posé en section 1.

Dans la méthode standard le nombre de degrés de liberté à prendre en compte est considérablement réduit par rapport à une méthode de Galerkin classique. D'autre part, la lacunarité effective de l'opérateur permet une comparaison réaliste avec les méthodes spectrales (où il n'y a pas de matrice d'opérateur à stocker). Une amélioration de la précision peut être apportée en préconditionnant la matrice par la diagonale. En effet, Beylkin [Beylkin93] pour la méthode BCR et Jaffard [Jaffard92] ont montré qu'alors, le conditionnement de l'opérateur était indépendant de l'ordre de la matrice, permettant la résolution de systèmes linéaires par méthodes itératives. Dans notre cas, les coefficients des opérateurs sont constants en temps, si δt est fixe. Une méthode directe (par inversion préalable de l'opérateur) est donc plus avantageuse. Le préconditionnement ici améliorerait la précision dans le calcul des petites échelles de la solution.

La résolution du problème (2) (Equations de Navier-Stokes bidimensionnelles périodiques) est actuellement en cours. L'utilisation de bases tensorielles d'ondelettes 2-D permet la factorisation de l'opérateur $(I - \Delta)$ et l'insertion directe de la méthode standard 1-D présentée dans cet article. Les conditions aux limites périodiques conduisent à un stockage économique (circulant et lacunaire) de $(I - \delta t \nu \Delta)^{-1}$. Le seul point encore délicat réside dans le codage (ligne par ligne et colonne par colonne) de la solution, si l'on veut tenir compte de sa lacunarité.

Remerciements :
Les auteurs tiennent à remercier C. Basdevant pour les nombreuses discussions qu'ils ont eues avec lui tout au long de la rédaction de cet article.

Bibliographie

[Bacry et al.91] Bacry (E.), Mallat (S.) et Papanicolaou (G.). – *A wawelet based space-time adaptive numerical method for partial differential equations.* – Rapport technique n° 591, Courant Institute of Mathematical Sciences, 1991.

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

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

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

[Beylkin92] Beylkin (G.). – 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] Beylkin (G.). – On wavelet-based algorithms for solving differential equations. – 1993. Soumis à CRC Press.

[Charton et al.95] Charton (Ph.) et Perrier (V.). – 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.

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

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

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

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

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

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

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

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

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

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

[Jaffard89] Jaffard (S.). – 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] Jaffard (S.). – Wavelet methods for fast resolution of elliptic problems. *SIAM J. on Numerical Analysis*, vol. 29, n° 4, 1992, pp. 965–986.

[Lazaar et al.94] Lazaar (S.), Liandrat (J.) et Tchamitchian (P.). – 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.

[Liandrat et al.92] Liandrat (J.), Perrier (V.) et Tchamitchian (P.). – 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, 1992.

[Maday et al.91] Maday (Y.), Perrier (V.) et Ravel (J. C.). – 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.

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

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

[Perrier et al.89] Perrier (V.) et Basdevant (C.). – 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.

A Calcul pratique des opérateurs

A.1 Ondelettes à support compact

La décomposition de l'opérateur $\frac{\partial}{\partial x}$ sur la base $(\varphi(x-k))_{k\in\mathbb{Z}}$, où φ est la fonction d'échelle d'un espace V_0 d'une analyse multirésolution, est exposée dans [Beylkin92], si φ est à support compact.
Le calcul des coefficients : $r_l = \int_{-\infty}^{+\infty} \varphi(x-l)\varphi'(x)\ dx$ est alors explicite. Les tableaux des valeurs de r_l figurent dans cet article (pour des ondelettes à support compact d'ordre 1 à 10).

A partir des coefficients r_l, on peut calculer les coefficients de l'opérateur $\frac{\partial}{\partial x}$ sur un espace V_p d'une analyse multirésolution périodique, à savoir :

$$s_i^p = \int_0^1 \varphi_0^p(x-\frac{i}{2^p}) \cdot \varphi_0'^p(x)\ dx \quad i = 0, 2^p-1 \tag{58}$$

On a le résultat suivant :

$$s_i^p = \sum_{l\in\mathbb{Z}} r_{2^p l - i} \quad i = 0, 2^p-1 \tag{59}$$

En effet,

$$s_i^p = \int_0^1 \varphi^p(x-\frac{i}{2^p}) \cdot \varphi^p(x)dx = 2^p \int_0^1 \sum_{k\in\mathbb{Z}} \varphi'(2^p(x-k)-i) \cdot \sum_{k'\in\mathbb{Z}} \varphi(2^p(x-k'))dx$$

Soit en posant $l = k'-k$:

$$\begin{aligned}
s_i^p &= 2^p \sum_{k\in\mathbb{Z}} \int_0^1 \sum_{l\in\mathbb{Z}} \varphi(2^p(x-k-l)-i) \cdot \varphi'(2^p(x-k))dx \\
&= 2^p \sum_{k\in\mathbb{Z}} \int_{-k}^{-k+1} \sum_{l\in\mathbb{Z}} \varphi(2^p(u-l)-i) \cdot \varphi'(2^p u)du \ ,\ u = x-k \\
&= 2^p \int_{\mathbb{R}} \sum_{l\in\mathbb{Z}} \varphi(2^p(u-l)-i) \cdot \varphi'(2^p u)du \\
&= \sum_{l\in\mathbb{Z}} \int_{\mathbb{R}} \varphi(v-2^p l-i) \cdot \varphi'(v)dv \ ,\ v = 2^p u
\end{aligned}$$

La décomposition de l'opérateur $\frac{\partial^2}{\partial x^2}$ peut ensuite être approchée par $(\Pi_{V_p}(\frac{\partial}{\partial x})\Pi_{V_p})(\Pi_{V_p}(\frac{\partial}{\partial x})\Pi_{V_p})$.

A.2 Ondelettes splines

Dans le cas où l'espace V_p est un espace de fonctions splines périodiques d'ordre m (c'est à dire des polynômes par morceaux de degré $m-1$, de classe $\mathcal{C}^{m-2}$), on peut calculer directement les coefficients $(s_i^p)_{i=0,2^p-1}$.

En posant :

$$\tau_m^p(k) = \left(\sum_{l\in\mathbb{Z}} \frac{k^m}{(k+l\ 2^p)^m}\right)^{-1}$$

Il vient :

$$\widehat{s}_q^p = \frac{2i\pi q}{2^p} \frac{\tau_{2m}^p(q)}{\tau_{2m-1}^p(q)} \quad q = 0, \cdots, 2^p-1$$

$$\text{où } \widehat{s}_q^p = \sum_{n=0}^{2^p-1} s_n^p e^{-\dfrac{2i\pi nq}{2^p}}$$

<u>Rappel</u> : pour $k \in \mathbb{Z}$, les coefficients de Fourier de $\varphi_0^p(x)$ sont donnés par : $\widehat{\varphi}_0(k) = \frac{1}{2^{\frac{p}{2}}\sqrt{\tau_{2m}^p(k)}}$. Un calcul analogue conduit à :

$$\widehat{t}_q^p = -\frac{4\pi^2 q^2}{2^p}\frac{\tau_{2m}^p(q)}{\tau_{2m-2}^p(q)}$$

$$\text{si } t_q^p = \int_0^1 \varphi_0(x - q2^{-p})\varphi_0''^p(x)\,dx$$

B 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 :

$$\widehat{a}_k = \frac{1}{N}\sum_{j=0}^{N-1} e^{i\frac{-2\pi jk}{N}} a_j \tag{60}$$

$$\text{Alors } \widehat{b}_k = \begin{cases} \frac{1}{\widehat{a}_k} & \text{si} \quad a_k \neq 0 \\ 0 & \text{si} \quad a_k = 0 \end{cases} \tag{61}$$

$$\text{Soit encore } b_j = \sum_{k=0}^{N-1} \frac{1}{\widehat{a}_k} e^{i\frac{2\pi jk}{N}} \tag{62}$$