



**HAL**  
open science

# Algorithme rapide et adaptatif pour le problème inverse de milieu en élastodynamique 3-D, avec fréquences et sources multiples

Stéphanie Chaillat, George Biros

## ► To cite this version:

Stéphanie Chaillat, George Biros. Algorithme rapide et adaptatif pour le problème inverse de milieu en élastodynamique 3-D, avec fréquences et sources multiples. 10e colloque national en calcul des structures, May 2011, Giens, France. pp.Clé USB. hal-00592791

**HAL Id: hal-00592791**

**<https://hal.science/hal-00592791>**

Submitted on 3 May 2011

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

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

# Algorithme rapide et adaptatif pour le problème inverse de milieu pour l'équation des ondes scalaire, avec fréquences et sources multiples

S. Chaillat<sup>1</sup>, G. Biros<sup>2</sup>

<sup>1</sup>Laboratoire POems, 32, Boulevard Victor, 75739 PARIS Cedex 15, France, stephanie.chaillat@ensta-paristech.fr

<sup>2</sup>College of Computing, Georgia Tech, 266 Ferst Drive, Atlanta, GA 30332-0765, USA, biros@gatech.edu

**Résumé** — On considère le problème d'inversion de l'équation des ondes scalaire dans un milieu 3-D. Le problème direct est formulé sous la forme d'une équation intégrale en supposant que les perturbations du milieu de référence sont petites. La résolution du problème inverse associé, en calculant la décomposition en valeurs singulières de la matrice du système, aurait un coût prohibitif. On propose un algorithme rapide pour diminuer drastiquement le nombre d'opérations. Dans une deuxième partie, on développe une méthode adaptative pour optimiser le nombre de points de la discrétisation spatiale.

**Mots clés** — problème inverse, algorithme adaptatif, équation des ondes scalaire.

## 1 Introduction

On propose un algorithme numérique pour la résolution du problème inverse de milieu. En particulier, on considère l'équation des ondes scalaire, sous l'approximation de Born. Etant donnés  $N_s N_\omega$  champs incidents ("d'illumination")  $\{u(\mathbf{r}; s, \omega)\}_{s=1, \omega=1}^{N_s, N_\omega}$  (où  $s$  correspond à une position spatiale de la source et  $\omega$  correspond à sa fréquence), le champ diffracté est mesuré  $\phi(\mathbf{r}_d; s, \omega)$  en  $N_d$  détecteurs  $\{\mathbf{r}_d\}_{d=1}^{N_d}$ . Le problème inverse consiste alors à retrouver la perturbation du milieu  $\eta(\mathbf{r})$  en résolvant

$$\phi(\mathbf{r}_d; s, \omega) = \int_H G(\mathbf{r}_d, \mathbf{r}; \omega) \eta(\mathbf{r}) u(\mathbf{r}; s, \omega) d\mathbf{r} \quad \text{pour } \eta. \quad (1)$$

Il s'agit d'une équation de type Lippmann-Schwinger, où  $G(\cdot, \cdot; \omega)$  est la fonction de Green pour le milieu de référence à la fréquence  $\omega$ ,  $H$  est le support du milieu de référence, et  $\mathbf{r}$  est un point de  $H$ . Après discrétisation avec  $N$  points de quadrature, on obtient

$$\phi(\mathbf{r}_d; s, \omega) = \sum_{j=1}^N G(\mathbf{r}_d, \mathbf{r}_j; \omega) \eta(\mathbf{r}_j) u(\mathbf{r}_j; s, \omega). \quad (2)$$

Dans la suite de cette communication,  $\phi$  est le champ diffracté généré par des diffracteurs ponctuels positionnés aux points de discrétisation  $\{\mathbf{r}_j\}_{j=1}^N$  et d'intensité  $\{\eta(\mathbf{r}_j)\}_{j=1}^N$ . La Figure 1 résume la configuration traitée.

Comme (2) est linéaire en  $\eta$ , on peut l'écrire sous la forme

$$\mathbf{M}\eta = \phi.$$

$\mathbf{M}$  est un opérateur dense. On souhaite inverser ce système en utilisant un algorithme de décomposition en valeurs singulières (SVD) pour résoudre

$$\min_{\eta} \|\mathbf{M}\eta - \phi\|_2.$$

Une approche, très coûteuse en temps et en mémoire serait d'assembler  $\mathbf{M}$  et d'appliquer ensuite une factorisation SVD classique. La complexité d'une telle approche est  $[\min(N_s N_\omega N_d, N)]^2 \times \max(N_s N_\omega N_d, N)$  [7]. Une autre approche serait d'utiliser une méthode itérative [1, 11], mais ces approches nécessitent la définition de bons préconditionneurs pour être envisageables.

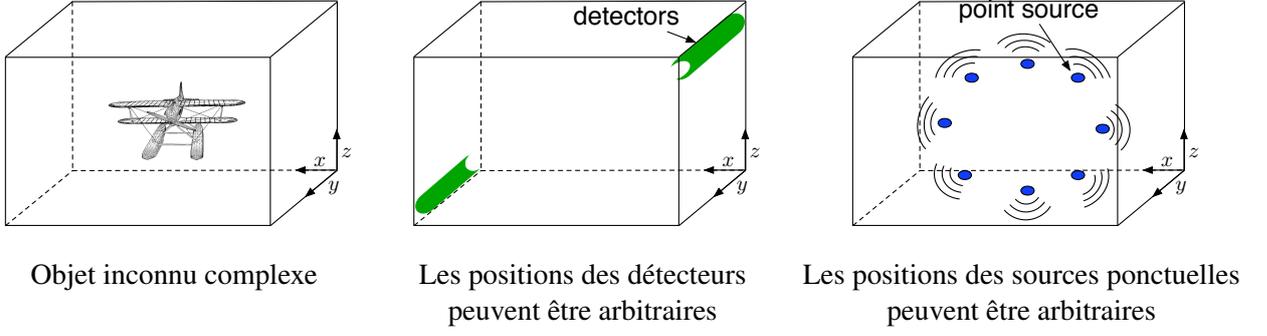


FIGURE 1 – On représente la perturbation du milieu par un ensemble de diffracteurs ponctuels dans un domaine 3-D,  $H$ . Les données sont constituées de mesures des champs diffractés dus aux différents champs incidents. On génère ces champs par des sources ponctuelles qui illuminent la région d'intérêt avec plusieurs fréquences. Les positions des détecteurs sont fixes et peuvent être arbitraires.

Notre première contribution est la construction d'une SVD approchée pour  $\mathbf{M}$ . Cette méthode est basée sur 3 algorithmes :

- une méthode de factorisation aléatoire qui permet de déterminer le rang d'une matrice. Il s'agit principalement de l'algorithme proposé dans [10] ;
- le pré-traitement du champ incident  $u$ , en utilisant la SVD, afin de réduire la dimension de  $N_s$  ;
- et une SVD récursive permet de calculer la SVD de  $\mathbf{M} = [\mathbf{M}_1 \ \mathbf{M}_2]^t$  à partir des SVDs de  $\mathbf{M}_1$  et  $\mathbf{M}_2$ .

On propose ainsi une factorisation SVD approchée de  $\mathbf{M}$  dont le nombre total d'opérations est  $O(N_\omega N N_d + N_\omega N N_s)$ . Cet algorithme est mise en oeuvre numériquement dans FaIMS.

Notre seconde contribution est la définition d'un algorithme d'adaptation en espace basé sur :

- l'utilisation d'une structure d'octree pour définir les régions du domaine à raffiner ;
- la définition d'un critère de raffinement. On étend la méthode utilisée pour détecter les sauts dans les fonctions régulières par morceaux, à partir de leurs données spectrales [3].

Dans la Section 2, on formule le problème. Dans la Section 3, on présente le principe de l'algorithme FaIMS, ainsi que l'algorithme de SVD récursive et quelques résultats numériques. Dans la Section 4, on définit l'algorithme d'adaptation en espace. On présente également les premiers résultats montrant l'efficacité de la méthode.

## 2 Formulation du problème inverse

On considère l'équation des ondes scalaire en régime harmonique

$$\nabla^2 v(\mathbf{r}) + k^2(\mathbf{r})v(\mathbf{r}) = -s(\mathbf{r}), \quad (3)$$

où  $s$  est le terme source et  $k$  est le nombre d'onde défini par  $k^2(\mathbf{r}) = \omega^2 c^{-2}(\mathbf{r})$ . On considère que  $k$  est donné par  $k^2(\mathbf{r}) = k_0^2 + \eta(\mathbf{r})$ , où  $k_0^2 = \omega^2 c_0^{-2}$  est le paramètre caractérisant le milieu de référence et  $\eta$  est la perturbation inconnue de ce milieu de référence. On note le champ total comme la somme du champ incident  $u$  et du champ diffracté  $\phi$ ,  $v = \phi + u$ . L'équation (3) devient

$$\nabla^2 \phi(\mathbf{r}) + k_0^2 \phi(\mathbf{r}) = -\eta(\mathbf{r})(\phi(\mathbf{r}) + u(\mathbf{r})). \quad (4)$$

En utilisant l'approximation de Born, on néglige le terme  $-\eta(\mathbf{r})\phi(\mathbf{r})$  et on obtient

$$\nabla^2 \phi(\mathbf{r}) + k_0^2 \phi(\mathbf{r}) = -\eta(\mathbf{r})u(\mathbf{r}). \quad (5)$$

La solution de (5) est donnée par [2] :

$$\phi(\mathbf{r}_d; s, \omega) = \int_H G(\mathbf{r}_d, \mathbf{r}; \omega) \eta(\mathbf{r}) u(\mathbf{r}; s, \omega) d^3 \mathbf{r}, \quad (6)$$

où  $G$  est la fonction de Green de l'espace infini donnée par

$$G(\mathbf{r}, \mathbf{r}') = \frac{\exp(ik_0|\mathbf{r} - \mathbf{r}'|)}{4\pi|\mathbf{r} - \mathbf{r}'|}. \quad (7)$$

L'équation (6) correspond au problème direct dans lequel, connaissant  $\boldsymbol{\eta}$  on cherche à calculer  $\phi$ . Dans le problème inverse, on cherche à retrouver l'inhomogénéité  $\boldsymbol{\eta}$  ( $\in \mathbb{R}^N$ ) connaissant  $\phi(\mathbf{r}_d; s, \omega)$ , un ensemble de mesures générées par  $N_s N_\omega$  champs incidents connus et mesurés en  $N_d$  détecteurs.

Pour une source  $s$ , une fréquence  $\omega$  et un détecteur  $d$  donnés, l'équation (6) devient

$$\phi_{ds}^\omega = \sum_j \mathbf{G}_{dj}^\omega \mathbf{u}_{js}^\omega \eta_j,$$

que l'on écrit sous la forme matricielle  $\phi = \mathbf{M}\boldsymbol{\eta}$ . Pour inverser pour  $\boldsymbol{\eta}$ , on calcule d'abord la SVD de  $\mathbf{M}$ , et on en déduit ensuite le pseudo-inverse de  $\mathbf{M}$ .

### 3 FaIMS

**Résumé de l'algorithme FaIMS.** L'algorithme FaIMS comporte quatre étapes principales.

- A : On transforme le système initial en un système équivalent mais de plus petite taille. Cette étape consiste en un pré-traitement du champ incident  $\mathbf{u}^\omega$  en utilisant la SVD aléatoire proposée par *Martinsson et al.* [10] pour réduire le nombre de sources de  $N_s$  à  $R_s$ .
- B : Pour une fréquence fixée, on calcule la SVD de la fonction de Green  $\mathbf{G}^\omega$  en utilisant la SVD aléatoire [10]. On approche ainsi chaque matrice  $\mathbf{G}^\omega \in \mathbb{R}^{N_d \times N}$  par une matrice de rang inférieur ou égal à  $R_g^\omega$ .
- C : Une fois les SVDs des fonctions de Green calculées, on utilise la SVD récursive pour les combiner, pour une fréquence fixée. On approche ainsi chaque matrice  $\mathbf{M}^\omega \in \mathbb{R}^{R_s R_g^\omega \times N}$  par une matrice de rang inférieur ou égal à  $R_\omega$ .
- D : En utilisant les résultats de l'étape C, on applique la SVD récursive pour obtenir la SVD de la matrice complète du système  $\mathbf{M}$ .

**SVD récursive.** On présente maintenant l'algorithme de SVD récursive. Soit  $\mathbf{M}$  une matrice de taille  $2m \times n$  et  $\mathbf{M}_1$  et  $\mathbf{M}_2$  deux matrices de taille  $m \times n$ . Le but est de calculer la SVD de  $\mathbf{M}$  donnée par

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 \\ \mathbf{M}_2 \end{bmatrix}, \quad (8)$$

à partir des SVDs de  $\mathbf{M}_1$  et  $\mathbf{M}_2$  :  $\mathbf{M}_1 = \Phi_1 \Lambda_1 \Psi_1^*$  et  $\mathbf{M}_2 = \Phi_2 \Lambda_2 \Psi_2^*$ . Lorsque des approximations faible rang de  $\mathbf{M}_1$  et  $\mathbf{M}_2$  sont possibles, cet algorithme est beaucoup plus rapide et moins coûteux en mémoire qu'une SVD classique. Soit  $r$  le rang de  $\mathbf{M}_1$  et  $\mathbf{M}_2$ . On cherche à déterminer  $\mathbf{U}$ ,  $\mathbf{V}$  et  $\Sigma$  telles que  $\mathbf{M} = \mathbf{U}\Sigma\mathbf{V}^*$ . On introduit une décomposition par blocs de  $\mathbf{U}$  et  $\Sigma$  :

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix} \text{ et } \Sigma = \begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \Sigma_2 \end{bmatrix}. \quad (9)$$

$\mathbf{U}_i$  est une matrice de taille  $m \times r$  et  $\Sigma_i$  est une matrice diagonale de taille  $r \times r$  ( $i = 1, 2$ ). La méthode procède en deux étapes. On détermine d'abord  $\mathbf{U}$  et  $\Sigma$ . On en déduit ensuite une approximation de  $\mathbf{V}$ .

Par définition, on sait que  $\mathbf{M}\mathbf{M}^*\mathbf{U} = \mathbf{U}\Sigma^2$ . De plus, en substituant les SVDs de  $\mathbf{M}_1$  et  $\mathbf{M}_2$  et (9) dans (8), on obtient le système

$$\begin{bmatrix} \Phi_1 & \mathbf{0} \\ \mathbf{0} & \Phi_2 \end{bmatrix} \begin{bmatrix} \Lambda_1^2 & \mathbf{Z}_1^* \mathbf{Z}_2 \\ \mathbf{Z}_2^* \mathbf{Z}_1 & \Lambda_2^2 \end{bmatrix} \begin{bmatrix} \Phi_1^* & \mathbf{0} \\ \mathbf{0} & \Phi_2^* \end{bmatrix} \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \end{bmatrix} \begin{bmatrix} \Sigma_1^2 & \mathbf{0} \\ \mathbf{0} & \Sigma_2^2 \end{bmatrix} \quad (10)$$

où  $\mathbf{Z}_i = \Psi_i \Lambda_i \in \mathbb{R}^{n \times r}$  ( $i = 1, 2$ ). Le changement de variable fondamental est d'introduire  $\mathbf{c}_k$  ( $k = 1, 2$ ) tel que

$$\mathbf{U}_k = \Phi_k \mathbf{c}_k. \quad (11)$$

Ensuite, en utilisant (11), (10) devient

$$\begin{bmatrix} \Lambda_1^2 & Z_1^* Z_2 \\ Z_2^* Z_1 & \Lambda_2^2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \begin{bmatrix} \Sigma_1^2 & \mathbf{0} \\ \mathbf{0} & \Sigma_2^2 \end{bmatrix}. \quad (12)$$

Si on définit  $\mathbf{A}$  telle que

$$\mathbf{A} = \begin{bmatrix} \Lambda_1^2 & Z_1^* Z_2 \\ Z_2^* Z_1 & \Lambda_2^2 \end{bmatrix}, \quad (13)$$

alors les vecteurs propres et valeurs propres de  $\mathbf{A}$  sont les vecteurs  $c_i$  et les coefficients  $\Sigma_i$  ( $i = 1, 2$ ).

En utilisant la SVD de  $\mathbf{M}$ , on sait que  $\mathbf{M}^* \mathbf{U} = \mathbf{V} \Sigma$ , ce qui en utilisant (8) et (11) est équivalent à

$$\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \mathbf{V} \Sigma.$$

L'expression finale de  $\mathbf{V}$  est donnée par

$$\mathbf{V}_j = \frac{1}{\Sigma_j} \sum_{k=1}^2 Z_k c_{k,j}, \quad 1 \leq j \leq 2r, \quad (14)$$

où  $\mathbf{V}_j \in \mathbb{R}^n$  est la  $j$ -ème colonne de  $\mathbf{V}$ . La complexité totale de cet algorithme est  $O(Rr(r+n+m))$ .

**Efficacité numérique de l'algorithme FaIMS.** Pour illustrer l'efficacité de notre algorithme d'inversion, on considère une géométrie d'avion biplan 3-D (modélisé par des diffracteurs ponctuels). L'avion mesure  $5\lambda$ . Il est englobé dans un domaine cubique  $H$  de côté  $L = 12\lambda$ . Ce cube est discrétisé avec une grille régulière comportant  $N = 51^3$  points. Les 12 sources et les 162 détecteurs sont positionnés sur une sphère (Fig. 2). Les données sont générées successivement à 8 fréquences. Sur la Figure 2c, on donne une vue 3-D de l'isosurface définie par  $\eta = 0.25 * \eta_{max}$ . L'inversion est assez précise. On peut clairement identifier les deux ailes et les deux flotteurs de l'avion. Le calcul direct de la SVD de la matrice globale du système ne serait pas possible aussi bien à cause des problèmes de mémoire que du nombre nécessaire d'opérations. Notre algorithme permet de traiter des problèmes à géométrie complexe, de manière précise, grâce à l'utilisation d'un grand nombre de données.

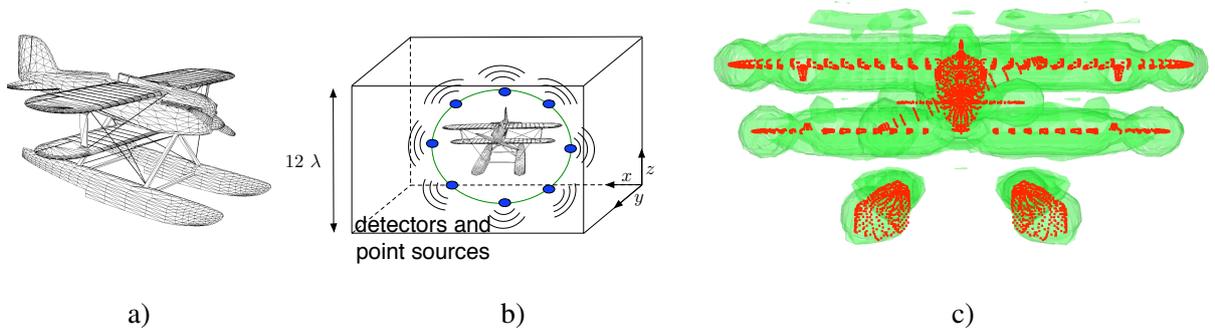


FIGURE 2 – Pour illustrer l'efficacité de notre algorithme d'inversion, on considère une géométrie d'avion biplan 3-D modélisé par des diffracteurs ponctuels. Les sources et les détecteurs sont positionnés sur une sphère. Les sous-figures a) et b) résument la configuration étudiée. La sous-figure c) représente une vue 3-D de l'isosurface définie par  $\eta = 0.25 * \eta_{max}$ .

## 4 Algorithme adaptatif en espace

**Résumé de l'algorithme.** On présente maintenant un algorithme d'adaptation, indépendamment de son intégration à FaMS. On utilise une structure d'octree pour guider l'algorithme d'adaptation. Cette méthode a deux avantages principaux. Tout d'abord il est ainsi plus simple de définir une quadrature précise pour calculer numériquement (6). Cette méthode permet aussi de s'affranchir d'une étape coûteuse de remaillage après chaque itération. Supposons qu'un critère pour définir les zones à raffiner a été défini. L'algorithme adaptatif procède comme suit :

- A : **Initialisation** du niveau 0 de l'arbre (tout le domaine de calcul). On définit une discrétisation à  $N$  points dans cette unique cellule.
- B : **Résolution du problème inverse** sur la grille définie à l'étape A.
- C : **Vérification du critère de raffinement.** Si il n'est pas nécessaire de raffiner, le calcul est terminé. Dans le cas contraire, on passe à l'étape D.
- D : **Subdivision** en 8 sous-cellules des cellules où la convergence n'est pas encore atteinte (voir Fig. 3) et définition d'une nouvelle discrétisation sur chaque sous-cellules.
- E : **Critère d'arrêt** : si le nombre maximal de niveaux (prédéfini en entrée du calcul) est atteint, on retourne à l'étape B. Sinon, le calcul est terminé.

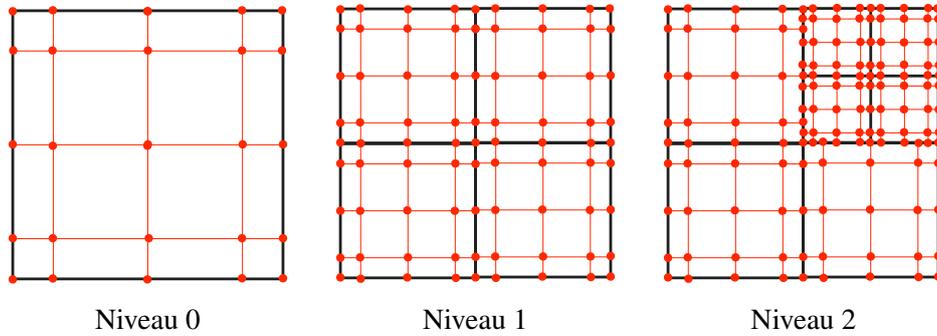


FIGURE 3 – Grilles 2-D pour 3 niveaux différents de l'octree. Chaque carré noir représente une cellule et chaque point rouge représente un point de discrétisation. Dans cet exemple, on utilise  $N = 5$  points de Chebyshev dans chaque direction. Le niveau 0 est la racine de l'octree (une seule cellule). Cette cellule est ensuite subdivisée en 4 cellules au niveau 1. Finalement, au niveau 2, une seule cellule est raffinée. Cette grille adaptée ne contient que 140 points au lieu de 289 si 16 cellules étaient utilisées. Le gain est évidemment encore plus important en 3D.

**Régularisation de Tikhonov.** Le problème étant mal-posé, il est nécessaire d'appliquer une méthode de régularisation. On utilise la méthode de Tikhonov [13]. On cherche à réduire les oscillations parasites dues à l'utilisation d'une grille non-uniforme. On doit résoudre

$$\underset{\eta}{\text{Minimiser}} \|\mathbf{M}\eta - \phi\|_2 + \beta \|\mathbf{D}\eta\|_2, \quad (15)$$

où  $\beta > 0$  est le paramètre de régularisation et  $\mathbf{D}$  est l'opérateur de dérivation du premier ordre. (15) revient à résoudre

$$(\mathbf{M}^*\mathbf{M} + \beta\mathbf{D}^*\mathbf{D})\eta = \mathbf{M}^*\phi. \quad (16)$$

On utilise la méthode L-curve [8, 9] pour déterminer la valeur optimale du paramètre de régularisation  $\beta$ . Cette méthode consiste en une représentation paramétrique de la norme du résidu  $\|\mathbf{M}\eta - \phi\|_2$  en fonction de la norme de la solution régularisée  $\|\mathbf{D}\eta\|_2$ , pour un certain nombre de valeurs du paramètre de régularisation  $\beta$ . Si  $\beta$  est trop grand, le résidu sera grand. D'un autre côté, si  $\beta$  est trop petit la solution sera trop oscillante. La courbe a une forme de L dont le coin donne la valeur de  $\beta$  qui minimise les deux normes.

**Critère de raffinement.** Le critère de raffinement est basé sur la méthode utilisée pour la détection de la localisation et de l'amplitude de sauts dans des fonctions régulières par morceaux, à partir de leur données spectrales. La méthode a été proposée par Gelb et Tadmor dans [3] et améliorée dans [4, 5, 6].

Nous rappelons rapidement la méthode pour la détection des sauts avant de préciser le critère de raffinement proposé. Soit une fonction régulière par morceaux  $\eta$  composée de  $m$  sauts de discontinuité aux abscisses  $x_j$ . Pour détecter les sauts, il est possible de regarder la dérivée de la projection spectrale de la fonction  $\eta$ , i.e. la fonction  $K_N\eta(x)$  définie par

$$K_N\eta(x) = \frac{\pi\sqrt{1-x^2}}{N} \sum_{k=0}^N \mu\left(\frac{k}{N}\right) \hat{\eta}_k \frac{d}{dx} T_k(x), \quad x \in [-1, 1], \quad (17)$$

où  $\mu \in C^2[0, 1]$  est appelé le facteur de concentration,  $\hat{\eta}_k$  est la  $k$ -ème composante de l'interpolation de Chebyshev de  $\eta$ ,  $T_k(x)$  est le polynôme de Chebyshev du premier type et  $N$  est l'ordre de l'interpolation de Chebyshev. On utilise une interpolation de Chebyshev aux points de Chebyshev à cause du caractère non-périodique des fonctions étudiées.

On peut montrer que la fonction  $K_N\eta(x)$  converge vers la position et l'amplitude des discontinuités. Cette propriété est donc utilisée pour définir le critère de raffinement. Dans une cellule de l'octree, si il existe un point de discrétisation où  $K_N\eta(x)$  est non nul, alors il y a une discontinuité dans cette cellule. Par conséquent, il est nécessaire de marquer cette cellule comme étant à raffiner.

**Premiers résultats de l'algorithme adaptatif sur un objet 1-D continu.** On considère un objet continu 1-D de taille  $\lambda$  dans un domaine 3-D cubique de côté  $L = 500\lambda$  (Fig. 4). On génère le champ incident avec deux sources excitées successivement à 4 fréquences. On mesure les données sur le plan défini par  $z = L$ . On discrétise chaque cellule avec  $N = 5$  points de Chebyshev.

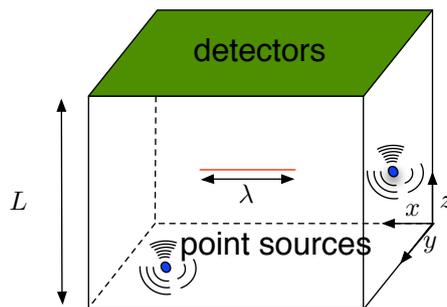


FIGURE 4 – On considère un objet continu 1-D de taille  $\lambda$  dans un domaine 3-D cubique de côté  $L = 500\lambda$ . On génère le champ incident avec deux sources excitées successivement à 4 fréquences. On mesure les données sur le plan défini par  $z = L$ .

La Figure 5 représente  $\eta$  pour deux itérations différentes de l'algorithme adaptatif : aux niveaux 2 et 7. Il suffit de seulement  $N = 41$  points au niveau 7 pour retrouver la position et la forme de l'inhomogénéité. La Figure 6 représente  $\eta$  dans le cas d'une grille uniforme de 50 ou 200 points. Même avec  $N = 200$  points, on ne retrouve pas avec précision la forme de l'inhomogénéité. On voit bien l'efficacité de l'utilisation d'un algorithme adaptatif. Finalement, on considère des données bruitées ( $\phi_{\text{bruitée}} = \phi + 0.01\|\phi\|_2 \times \text{Gaussienne}$ ). La Figure 7 représente  $\eta$  pour deux itérations différentes de l'algorithme adaptatif : aux niveaux 3 et 8. Il suffit, comme dans le cas sans bruit, de seulement  $N = 41$  points au niveau 8 pour retrouver la position et la forme de l'inhomogénéité.

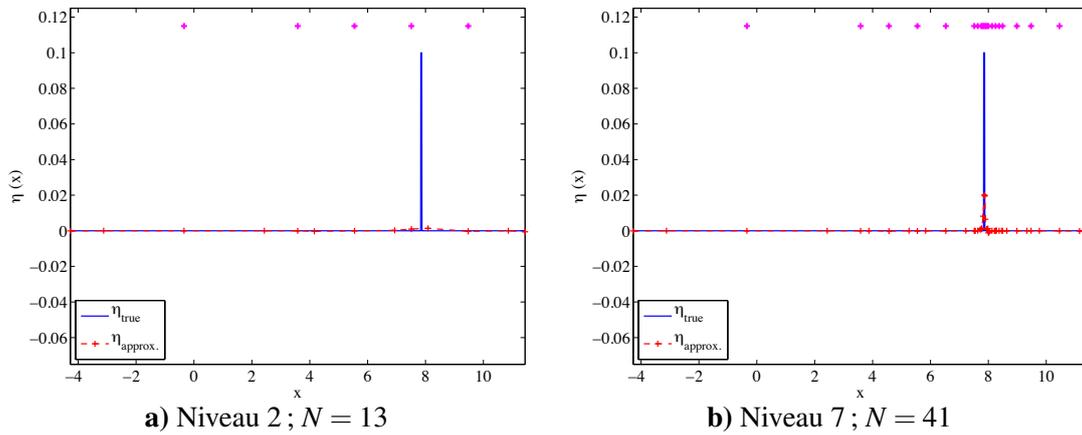


FIGURE 5 – Valeur de  $\eta$  pour deux itérations différentes de l’algorithme adaptatif (en bleu la valeur réelle et en rouge la valeur calculée). Avec seulement  $N = 41$  points au niveau 7, on retrouve la position et la forme de l’inhomogénéité.

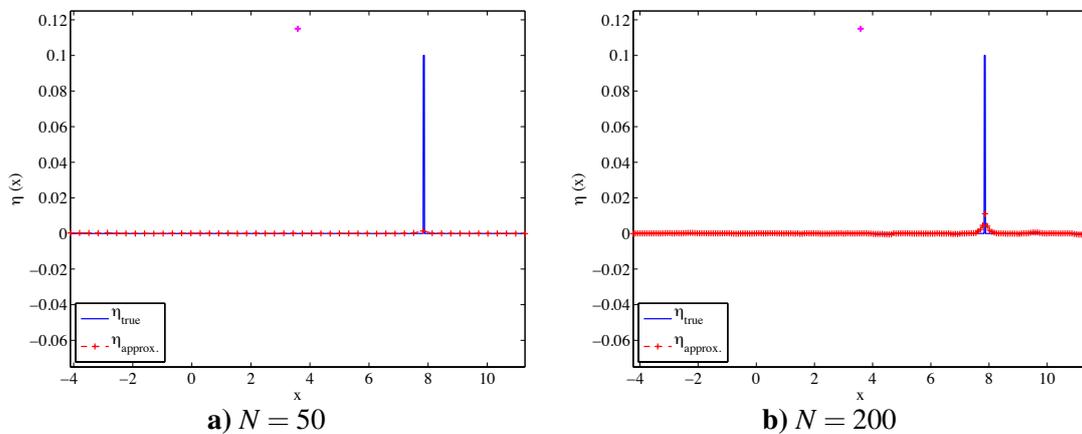


FIGURE 6 – Valeur de  $\eta$  (en bleu la valeur réelle et en rouge la valeur calculée) si on utilise une grille uniforme (pour deux valeurs de discrétisation). On remarque que la position mais pas la forme de l’inhomogénéité est retrouvée avec  $N = 200$  alors que seulement  $N = 41$  points sont nécessaires avec l’algorithme adaptatif (Fig. 5b).

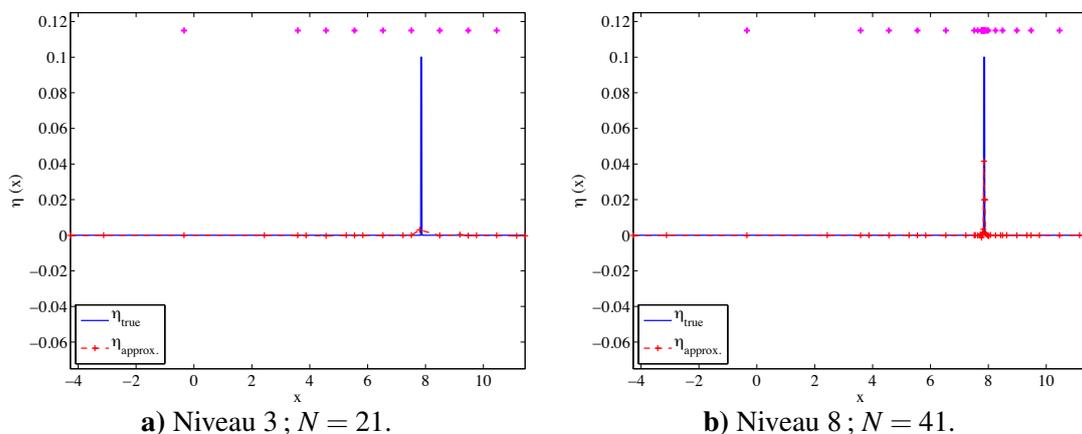


FIGURE 7 – Valeur de  $\eta$  pour deux itérations différentes de l’algorithme adaptatif en présence de bruit gaussien (en bleu la valeur réelle et en rouge la valeur calculée). Avec seulement  $N = 41$  points au niveau 8, on retrouve la position et la forme de l’inhomogénéité.

## Conclusion

On a présenté une méthode pour l'inversion de l'équation des ondes scalaire en régime harmonique. Cette méthode est mise en oeuvre numériquement dans FaIMS. L'idée principale est de tirer profit de la SVD aléatoire pour calculer les SVDs de sous-matrices de petites tailles. On applique ensuite une SVD récursive. Cette méthode de SVD récursive est la première contribution de ce travail. La complexité de la méthode d'inversion est ainsi drastiquement réduite par rapport à une SVD classique appliquée sur la matrice complète du système. Le principal avantage de cet algorithme est de permettre de positionner les sources et les détecteurs de manière complètement arbitraire et de prendre en compte des données générées à plusieurs fréquences. Le nombre élevé de données permet ainsi d'obtenir une meilleure précision de l'inversion. La seule hypothèse que l'on fait est que l'on peut effectuer l'approximation de Born.

Dans une deuxième partie, on a présenté une piste d'amélioration de la méthode. Il s'agit de développer un algorithme adaptatif en espace. Cette méthode est basée sur l'utilisation d'un octree pour diriger le raffinement de certaines régions du domaine de calcul, et sur la définition d'un critère de raffinement efficace. Ce travail n'est pas pour l'instant mis en oeuvre dans FaIMS mais sera effectué prochainement. Il faut dans un premier temps tester l'efficacité de la méthode sur des problèmes 3-D. Ensuite, comme il n'est plus possible d'inverser directement le système (16) avec la SVD calculée par FaIMS, on utilisera l'algorithme pour proposer un préconditionnement efficace pour les méthodes itératives. La méthode sera aussi parallélisée pour traiter des problèmes de taille plus importante.

## Références

- [1] J. Chung, J.G. Nagy, and D.P. O'Leary. *A Weighted-GCV Method for Lanczos-Hybrid Regularization*, Electronic Transactions on Numerical Analysis, vol. 28, pp. 149–167, 2008.
- [2] R. Courant and D. Hilbert. *Methods of Mathematical Physics, vol. II*, Interscience Publishers, 1961.
- [3] A. Gelb and E. Tadmor. *Detection of edges in spectral data*, Applied and Computational Harmonic Analysis, vol. 7, pp. 101–135, 1999.
- [4] A. Gelb and E. Tadmor. *Detection of edges in spectral data II. Nonlinear enhancement*, SIAM Journal on Numerical Analysis, vol. 38, pp. 1389–1408, 2001.
- [5] A. Gelb and E. Tadmor. *Spectral reconstruction of piecewise smooth functions from their discrete data*, M2AN, vol. 36, pp. 155–175, 2002.
- [6] A. Gelb and E. Tadmor. *Adaptive edge detection for piecewise smooth data based on the minmod limiter*, Journal of Scientific Computing, vol. 28, pp. 279–306, 2006.
- [7] G.H. Golub and C.H. Van Loan. *Matrix Computations*, Johns Hopkins, third edition, 1996.
- [8] P.C. Hansen. *Analysis of discrete ill-posed problems by means of the l-curve*, SIAM Review, vol. 34, pp. 561–580, 1992.
- [9] P.C. Hansen. *The l-curve and its use in the numerical treatment of inverse problems*, in Computational inverse problems in electrocardiology, P. Johnston, Ed., pp. 119–142. WIT Press, Southampton, 2001.
- [10] P.G. Martinsson, V. Rokhlin, and M. Tygert. *A randomized algorithm for the approximation of matrices*, Tech. Rep. YALEU/DCS/RR-1361, Yale CS, 2006.
- [11] C.C. Paige and M.A. Saunders. *Algorithm 583 : LSQR : Sparse linear equations and least squares problems*, ACM Transactions on Mathematical Software (TOMS), vol. 8, no. 2, pp. 195–209, 1982.
- [12] J.C. Schotland and V.A. Markel. *Inverse scattering with diffusing waves*, J. Opt. Soc. Am. A., vol. 18, pp. 2767–2777, 2001.
- [13] A.N. Tikhonov, *Solution of incorrectly formulated problems and the regularization method*, Soviet Math. Dokl., vol. 5, pp. 1035–1038, 1963.
- [14] L.N. Trefethen. *Spectral methods in Matlab*, SIAM, 2000.
- [15] C.R. Vogel. *Computational methods for inverse problems*, SIAM frontiers in applied mathematics, 2002.