

Université du Québec
Institut national de la recherche scientifique
Centre Eau Terre Environnement

Inversion des formes d'ondes complètes viscoélastique

Par
Gabriel Fabien-Ouellet

Thèse présentée pour l'obtention du grade de
Philosophiæ Doctor, Ph.D.
en sciences de la Terre

Jury d'évaluation

| | |
|---------------------------|--|
| Examineur externe | Jean Virieux Université Joseph Fourier |
| Examineur externe | Laurent Demanet Massachusetts Institute of Technology |
| Examineur interne | Mathieu Duchesne Commission géologique du Canada |
| Directeur de recherche | Erwan Gloaguen INRS-ETE |
| Co-directeur de recherche | Bernard Giroux INRS-ETE |

Remerciements

J'aimerais tout d'abord remercier mon directeur de thèse, Erwan Gloaguen, qui m'a convaincu d'entreprendre cette folle aventure qu'est le doctorat. Merci de m'avoir poussé à me dépasser et de m'avoir supporté dans mes projets scientifiques parfois plus qu'ambitieux. Je n'aurais jamais pu accomplir autant sans ton support.

Merci aussi à mon codirecteur Bernard Giroux, pour les nombreuses discussions sur la sismique et la programmation, en conférence ou autour d'une bière! J'ai grandement apprécié tes conseils au cours de mes années à l'INRS.

Merci aux membres du jury, Jean Virieux, Laurent Demanet et Mathieu Duchesne, d'avoir accepté de porter un regard critique sur mes travaux. Mon travail n'en sera que meilleur.

J'aimerais remercier aussi Mathieu pour toutes les discussions qu'on a eu sur le monde de la géophysique, sur la sismique ou sur le vélo! J'ai grandement apprécié collaborer avec toi!

Je tiens aussi à remercier tous mes collègues de bureau, Martin, Marc, Véronique, Pierre, Simon, Jean-Sébastien, Lorenzo, Patrick et Olivier. Ces années en votre compagnie furent des plus agréables. Merci en particulier à Martin, pour ces parties de basket l'été, ton franc-parler et la saine compétition entre nos projets entrepreneuriaux.

Un grand merci aussi à Lorenzo pour m'avoir permis d'utiliser le modèle géologique qu'il a développé pour la séquestration géologique du CO₂.

Je dois aussi remercier mon partenaire dans Thermoroc, Alain Zubrzycki, qui m'a permis de garder le contact avec la réalité du terrain au cours de ces années de doctorat.

Enfin merci à mes parents, qui m'ont toujours supporté inconditionnellement dans mes projets. Merci à Jeanne de m'avoir accompagné en cette dernière année de doctorat, qui fut, à tout le moins, étourdissante!

Résumé

L'imagerie non destructive de l'intérieur d'un solide est un sujet de grand intérêt en sciences de la Terre, et ce, à une variété d'échelles. Le sismologue voulant étudier la structure du manteau terrestre, le géophysicien d'exploration cherchant un nouveau gisement, l'hydrogéologue souhaitant modéliser l'écoulement de contaminants dans un aquifère; tous ont besoin d'estimer la structure et les propriétés physiques du sol. Une des méthodes les plus avancées pour étudier l'intérieur de la sous-surface a été développée au courant des dernières décennies: l'inversion de formes d'ondes complètes. L'engouement envers cette technique provient du fait qu'elle permettrait, à terme, de reconstruire de multiples propriétés mécaniques simultanément à une résolution inégalée. Pour l'instant, l'inversion en formes d'onde complètes est utilisée surtout dans l'industrie pétrolière pour obtenir des modèles de vitesse des ondes de compression plus précis. L'inversion de multiples paramètres (propriétés des ondes de cisaillement, les facteurs d'atténuation, les coefficients d'anisotropie, etc.) reste marginale en raison de son coût élevé en calcul et des difficultés de convergence lors de l'inversion.

Cette thèse porte sur le développement d'une approche d'inversion de formes d'ondes complètes prenant en compte les phénomènes élastiques et les pertes visqueuses, c'est-à-dire l'inversion viscoélastique. La base de cette approche est le calcul du gradient par la méthode adjointe. Les équations adjointes sont ainsi développées pour la formulation vitesse-contrainte de l'équation d'onde, largement utilisée pour la modélisation sismique viscoélastique. Le deuxième élément de cette approche est la résolution numérique des équations directes et adjointes à l'aide de plateformes de calcul parallèles. En utilisant les divers degrés de parallélisme accessibles sur les plateformes de calcul modernes, il est possible d'accélérer significativement les calculs de l'inversion en formes d'onde complètes. Notamment, le calcul sur des cartes graphiques avec OpenCL mène à une accélération jusqu'à deux ordres de grandeur par rapport aux calculs sur un seul coeur d'un processeur standard. Le dernier élément de cette approche est une stratégie d'inversion permettant de réduire le couplage entre différentes classes de paramètres. Le niveau de résolution atteignable lors de la reconstruction des vitesses et des facteurs de qualité des ondes S et des ondes P avec une telle stratégie est illustré par une étude de cas synthétique dans le contexte du suivi sismique de l'injection du CO_2 .

En sommes, les résultats présentés dans cette thèse confirment qu'il est possible d'inclure et d'inverser les propriétés viscoélastiques d'un solide grâce à l'inversion de formes d'ondes complètes, et que l'utilisation efficace du calcul parallèle permet de mitiger le coût additionnel par rapport à une inversion purement acoustique. Ces résultats entrouvrent la voie à de nouvelles applications de cette technologie, hors du contexte traditionnel de l'exploration pétrolière.

Mots-clés Inversion de formes d'ondes complètes; viscoélasticité; calcul parallèle; OpenCL; méthode adjointe

Abstract

Non-destructive imaging of the internal structure of solids is a topic of interest in Earth sciences, at a variety of scales. The seismologist might want to study the structure of the mantle, the exploration geophysicist might want to find a new oil reservoir, the hydrogeologist might want to model the migration of contaminants in an aquifer. All of them need an estimate of the structure and of the physical properties of the subsurface. One of the most powerful techniques to study the interior of the Earth has been developed in the last decades: full waveform inversion. The excitement over this method spurs from the promise of a multiparameter imaging at unprecedented spatial resolution. For the moment, full waveform inversion is mostly used in the oil and gas industry to obtain accurate P -wave velocity models. The inversion of multiple parameters, such as shear wave properties and wave attenuation, remains marginal because of the high computational cost.

This thesis proposes a new approach for full waveform inversion that integrates elastic waves and viscous loss, that is, viscoelastic inversion. The first element of this approach is the computation of the misfit gradient. We obtain the adjoint equations adapted to the velocity-stress formulation of the wave equation, commonly used for seismic viscoelastic modelling. The second element of this approach is the use of the different levels of parallelism commonly found in modern computing architectures to solve the wave equation. Most notably, using graphical processing units with OpenCL leads to a speedup of nearly two orders of magnitude over single thread computing on a standard processor. The last element of this approach is an inversion strategy to reduce crosstalk between different parameter classes. The application of this strategy to a synthetic case study demonstrates the resolution that can be reached for P - and S -wave velocities and quality factors.

In summary, the results presented in this thesis confirm that it is possible to include and invert for viscoelastic effects with full waveform inversion, at a reasonable cost with the aggressive use of parallel computing. This brings new applications of this technology within closer reach, outside of the traditional context of oil and gas exploration.

Keywords Full waveform inversion; viscoelasticity; parallel computing; OpenCL; adjoint method

Table des matières

| | |
|---|-------------|
| Remerciements | iii |
| Résumé | v |
| Abstract | vii |
| Table des matières | ix |
| Liste des figures | xiii |
| Liste des tableaux | xvii |
| Liste des algorithmes | xix |
| Liste des symboles | xxi |
| 1 Introduction | 1 |
| 1.1 Vers une estimation quantitative des propriétés physiques | 2 |
| 1.2 Méthodes sismiques | 3 |
| 1.2.1 Les méthodes traditionnelles et leurs limites | 4 |
| 1.2.2 L'inversion de formes d'ondes complètes | 5 |
| 1.3 Problématique | 7 |
| 1.4 Objectifs de la thèse | 9 |
| 2 Survol théorique | 11 |
| 2.1 Propagation des ondes | 12 |
| 2.1.1 Élasticité linéaire | 12 |
| 2.1.2 Viscoélasticité linéaire | 14 |
| 2.1.3 Équation d'onde anisotrope viscoélastique | 17 |
| 2.1.4 Formulation du déplacement | 19 |
| 2.1.5 Formulation en fréquence | 19 |
| 2.1.6 Approximation isotrope | 20 |
| 2.1.7 Approximation élastique | 21 |
| 2.1.8 Approximation 2D | 21 |
| 2.1.9 Solution analytique | 22 |
| 2.2 Résolution par différences finies de l'équation d'onde | 23 |
| 2.2.1 Choix de la méthode | 23 |
| 2.2.2 Fondements théoriques | 25 |
| 2.2.3 Grilles décalées | 26 |

| | | |
|----------|--|------------|
| 2.2.4 | Dispersion et stabilité | 27 |
| 2.2.5 | Équation d'onde viscoélastique isotrope en 3D avec grilles décalées | 29 |
| 2.2.6 | Conditions aux limites | 36 |
| 2.3 | Inversion | 40 |
| 2.3.1 | Formulation du problème | 40 |
| 2.3.2 | Fondement des méthodes de recherche linéaire | 42 |
| 2.3.3 | Dimensions du problème d'inversion | 43 |
| 2.3.4 | Méthodes itératives de recherche linéaire | 45 |
| 2.3.5 | Le défi des minimums locaux | 52 |
| 3 | Méthode adjointe | 57 |
| 3.1 | Fondements de la méthode | 58 |
| 3.2 | Opérateurs auto-adjoints | 61 |
| 3.3 | Application au régime anisotrope viscoélastique | 63 |
| 3.4 | Équivalence de la formulation vitesse-contrainte et de la formulation en déplacement | 70 |
| 3.5 | Application au régime isotrope viscoélastique | 72 |
| 3.6 | Conclusion | 75 |
| 4 | Résolution par différences finies sur des plateformes de calcul hétérogènes | 77 |
| 4.1 | Parallélisme des différences finies | 78 |
| 4.2 | Plateformes de calculs hétérogènes | 80 |
| 4.3 | Calcul scientifique sur cartes graphiques | 82 |
| 4.4 | Le langage OpenCL | 85 |
| 4.5 | Implémentation parallèle avec MPI et OpenCL | 87 |
| 4.5.1 | Parallélisme des sources | 88 |
| 4.5.2 | Parallélisme des mises à jour | 89 |
| 4.5.3 | Parallélisme intra-processeur OpenCL | 90 |
| 4.5.4 | Calcul du gradient | 92 |
| 4.6 | Validation du code | 96 |
| 4.6.1 | Validation de la modélisation sismique | 96 |
| 4.6.2 | Validation du gradient de la fonction coût | 97 |
| 4.7 | Comparaison des performances | 99 |
| 4.7.1 | Accélération selon le type d'engin de calcul | 100 |
| 4.7.2 | Performance du calcul du gradient | 101 |
| 4.7.3 | Coût des schémas en différences finies d'ordre élevé | 102 |
| 4.7.4 | Scalabilité sur des grappes de calcul hétérogènes | 103 |
| 4.8 | Conclusion | 105 |
| 5 | Étude de cas: Suivi de la séquestration géologique du CO₂ | 107 |
| 5.1 | Problématique | 108 |
| 5.1.1 | Contexte géologique | 109 |
| 5.1.2 | Construction du modèle synthétique | 109 |
| 5.1.3 | Modélisation du transport de CO ₂ | 110 |
| 5.1.4 | Suivi sismique entre forages | 112 |
| 5.2 | Développement d'une stratégie d'inversion multiparamètres | 113 |
| 5.2.1 | Interdépendance des paramètres viscoélastique | 113 |
| 5.2.2 | Stratégie d'inversion | 119 |
| 5.3 | Suivi temporel du panache de CO ₂ | 120 |

| | | |
|----------|---|------------|
| 5.3.1 | Inversion du modèle avant injection | 120 |
| 5.3.2 | Inversion du modèle après injection | 125 |
| 5.3.3 | Suivi du CO ₂ | 127 |
| 5.4 | Conclusion | 129 |
| 6 | Conclusion et perspectives | 131 |
| | Références | 135 |

Liste des figures

| | | |
|------|--|----|
| 1.1 | Schéma de l'inversion de formes d'ondes complètes. | 6 |
| 2.1 | Diagramme de contrainte déformation d'un solide, montrant le régime élastique valable pour de faibles contraintes. | 12 |
| 2.2 | Directions et plans des contraintes appliquées à un volume élémentaire d'un solide. . | 13 |
| 2.3 | Diagramme de contrainte déformation d'un solide viscoélastique, montrant que les contraintes dépendent de l'historique de déformation. | 14 |
| 2.4 | Modèle du solide de Maxwell généralisé, consistant en une superposition en parallèle de plusieurs solides de Maxwell, constitués d'un ressort et un amortisseur en série. . | 15 |
| 2.5 | Profil d'atténuation en fréquence du solide de Maxwell généralisé pour plusieurs niveaux d'atténuation, avec un seul mécanisme (droite) et deux mécanismes (gauche). | 16 |
| 2.6 | Réponse en fréquence des opérateurs en différences finies: a) grille standard (coefficients éq. (2.40)), b) coefficients de Taylor en grille standard, c) grille décalée (coefficients éq. (2.41)). En d), le noyau exact de convolution est montré avec les points d'échantillonnage de la grille standard (cercles ouverts) et de la grille décalée (carrés pleins). | 28 |
| 2.7 | Position des dérivées spatiales avant et arrière pour un schéma en différences finies d'ordre 2. | 30 |
| 2.8 | Cellule élémentaire de la grille standard décalée utilisée pour discrétiser l'équation 3D viscoélastique isotrope. | 32 |
| 2.9 | Ondelette de référence (ligne continue) et ondelette modélisée (ligne pointillée) à la figure du haut. Lorsque la fonction coût est calculée en fonction de la distance Δt entre les deux ondelettes, la figure du bas est obtenue. | 53 |
| 2.10 | Fonction coût pour des fréquences décroissantes de a) vers d). | 54 |
| 3.1 | Schéma des contours $f(x)$ d'une fonction à minimiser le long d'une contrainte $g(x)$. | 59 |
| 3.2 | Comparaison numérique entre la forme traditionnelle (forme 1) de l'équation d'onde et la forme utilisant la variable de mémoire alternative (forme 2). | 65 |
| 4.1 | Loi d'Amdhal pour différentes portions parallèles d'une tâche. | 79 |
| 4.2 | Schématisme d'une grappe de calcul (gauche) et d'un noeud hétérogène de cette grappe (droite). | 81 |
| 4.3 | Architecture de von Neumann (haut), architecture conceptuelle d'un CPU à 4 coeurs (bas, gauche) et d'un GPU avec 20 coeurs (bas, droite). | 83 |
| 4.4 | Schéma général d'un processeur parallèle adopté par OpenCL et sa généralisation avec l'interface MPI. | 85 |
| 4.5 | Décomposition de domaine et protocoles de communication utilisés. | 90 |

| | | |
|------|---|-----|
| 4.6 | Schéma des cellules de la grille transférées de la mémoire globale à la mémoire locale à l'intérieur d'un groupe de travail local. Les cellules en blancs sont mises à jours tandis que les cellules en gris ne sont utilisées que pour les calculs des dérivées spatiales. | 93 |
| 4.7 | Comparaison entre la solution analytique et les résultats en différences finies obtenus avec SeisCL pour quatre niveaux d'atténuation. | 98 |
| 4.8 | Comparaison du gradient obtenu par différences finies et par le modèle adjoint. Le sources et les receveurs sont représentés respectivement par les triangles et les cercles rouges. | 99 |
| 4.9 | Accélération obtenue avec différents types d'engins de calcul par rapport à une implémentation en série pour le cas 3D (haut) et 2D (bas). | 101 |
| 4.10 | Ralentissement mesuré pour le calcul du gradient par sélection de fréquences pour un nombre de fréquences croissant. | 102 |
| 4.11 | Ralentissement par rapport à un schéma en différences finies d'ordre 2 pour les schémas d'ordre supérieurs. | 104 |
| 4.12 | Test de scalabilité forte pour un modèle de 96x96x1000. | 105 |
| 5.1 | Le modèle géologique montrant le réservoir ciblé pour le CO ₂ en a) et une diagraphie sismique représentative de la région en b). Les sources sismiques et les receveurs sont respectivement représentés par des triangles et des carrés noirs. | 110 |
| 5.2 | Modèle avant injection des 4 paramètres sismiques que l'on veut imager par inversion. | 111 |
| 5.3 | Les saturations en CO ₂ après 15 ans d'injection (gauche), les vitesses (centre) et le niveau d'atténuation (droite) des ondes P suite à l'injection. | 112 |
| 5.4 | Effet du couplage entre V_p (rangée du haut) et τ_p (rangée du bas) pour une source explosive. Position des perturbations (colonne de gauche) et gradients normalisés pour un faible $\Delta\tau_p$ de 0.01 (50 %) (colonne du centre) et un fort $\Delta\tau_p$ de 0.06 (300 %). Les sources sont en rouge et les receveurs sont en jaune. | 115 |
| 5.5 | Effet du couplage entre V_s (rangée du haut) et τ_s (rangée du bas) pour une source orientée en z . Position des perturbations (colonne de gauche) et gradients normalisés pour un faible $\Delta\tau_s$ de 0.01 (50 %) (colonne du centre) et un fort $\Delta\tau_s$ de 0.06 (300 %). Les sources sont en rouge et les receveurs sont en jaune. | 116 |
| 5.6 | Effet du couplage entre V_p (rangée du haut) et V_s (rangée du bas) pour une source orientée en z . Position des perturbations (colonne de gauche) et gradients normalisés pour un faible ΔV_p de 50 m/s (1.4%) (colonne du centre) et un fort ΔV_p de 200 m/s (8 %). Les sources sont en rouge et les receveurs sont en jaune. | 117 |
| 5.7 | Effet du couplage entre V_p (rangée du haut) et V_s (rangée du bas) pour une source explosive. Position des perturbations (colonne de gauche) et gradients normalisés pour un faible ΔV_s de 50 m/s (2.5%) (colonne du centre) et un fort ΔV_s de 100 m/s (5 %). Les sources sont en rouge et les receveurs sont en jaune. | 118 |
| 5.8 | Regroupement à déport nul pour le vrai modèle avant injection (première colonne), le regroupement à déport nul et les résidus du modèle initial (deuxième et troisième colonnes) et le regroupement à déport nul et les résidus du modèle inversé (quatrième et cinquième colonnes). Les première et deuxième rangées présentent les composantes v_z et v_x de la source orientée en z , tandis que les troisième et quatrième rangées présentent les composantes v_z et v_x de la source explosive. Les lignes bleues et rouge indiquent les premières arrivées des ondes P et S , respectivement. | 122 |

| | | |
|------|---|-----|
| 5.9 | Le modèle vrai avant injection (première colonne), le modèle initial (deuxième colonne) et le modèle inversé (troisième colonne). La vitesse des ondes P est présentée à la première rangée, la vitesse des ondes S à la deuxième rangée et leurs niveaux d'atténuation aux troisième et quatrième rangées. | 123 |
| 5.10 | Graphique quantile-quantile entre le modèle vrai avant injection et les modèles initiaux et inversés. Les lignes rouges représentent une correspondance exacte entre les deux distributions et les points noirs représentent les quantiles calculés. | 124 |
| 5.11 | Le modèle après l'injection de CO_2 : modèle vrai (première colonne), modèle initial (deuxième colonne) et modèle inversé (troisième colonne). La vitesse des ondes P est présentée à la première rangée et le niveau d'atténuation est présenté à la deuxième rangée. | 126 |
| 5.12 | Regroupement à déport nul pour le vrai modèle après injection (première colonne), le regroupement à déport nul et les résidus du modèle initial (deuxième et troisième colonnes) et le regroupement à déport nul et les résidus du modèle inversé (quatrième et cinquième colonnes). Les première et deuxième rangées présentent les composantes v_z et v_x de la source explosive. Les lignes bleues pleines et hachurées indiquent respectivement des premières arrivées des ondes P avant et après injection et les lignes pleines rouges indiquent les premières arrivées S | 127 |
| 5.13 | Graphique quantile-quantile entre le modèle vrai après injection et les modèles initiaux et inversés. Les lignes rouges représentent une correspondance exacte entre les deux distributions et les points noirs représentent les quantiles calculés. | 128 |
| 5.14 | La saturation en CO_2 après 15 ans d'injection (première colonne), le changement en V_p (deuxième colonne) et du facteur de qualité Q_p (troisième colonne) entre le modèle avant et après injection. La première rangée montre le modèle vrai et la deuxième montre les changements obtenus de l'inversion. | 129 |

Liste des tableaux

| | | |
|-----|--|-----|
| 5.1 | Résumé des pairs de paramètres pour lesquels le couplage est étudié. | 114 |
|-----|--|-----|

Liste des algorithmes

| | | |
|----|---|-----|
| 1 | Méthode de la descente la plus forte | 45 |
| 2 | Méthode du gradient conjugué non linéaire | 47 |
| 3 | Méthode de Newton tronquée | 49 |
| 4 | Méthode de l-BFGS | 50 |
| 5 | Méthode du gradient stochastique | 51 |
| 6 | Recherche linéaire à rebours | 52 |
| 7 | SeisCL: Routine générale | 88 |
| 8 | Routine de propagation | 91 |
| 9 | <i>Kernel</i> d'actualisation des contraintes et des variables de mémoire | 92 |
| 10 | Stratégie de la propagation arrière | 95 |
| 11 | Stratégie de la sélection des fréquences | 96 |
| 12 | Méthodologie d'inversion | 120 |

Liste des symboles

| | |
|---------------------------------------|---|
| α | Grandeur de pas |
| Δt | Pas temporel |
| Δx | Pas spatial en x |
| ϵ | Déformation |
| λ | Deuxième constante de Lamé |
| $\mathcal{F}(\mathbf{m})$ | Modèle direct |
| \mathcal{L} | Lagrangien |
| $\mathcal{X}(\mathbf{m}; \mathbf{d})$ | Fonction coût |
| μ | Module de cisaillement |
| $\nabla \mathcal{X}$ | Gradient de la fonction coût |
| $\nabla^2 \mathcal{X}$ | Hessienne de la fonction coût |
| ω | Fréquence angulaire |
| $\mathbf{0}_n$ | Matrice de zéros $n \times n$ |
| ϕ | Variable d'état |
| Ψ | Matrice de relaxation |
| ψ | Variable adjointe |
| \mathbf{d} | Données enregistrées |
| \mathbf{I}_n | Matrice identité $n \times n$ |
| \mathbf{m} | Paramètres d'inversion |
| \mathbf{p} | Direction de descente |
| \mathbf{S} | Opérateur d'échantillonnage |
| ρ | Densité |
| σ | Contrainte |
| τ | Niveau d'atténuation |
| $\tau_{\epsilon l}$ | Temps de relaxation des déformations du mécanisme de relaxation l |
| $\tau_{\sigma l}$ | Temps de relaxation des contraintes du mécanisme de relaxation l |

| | |
|-------------------|---|
| c_{ijkl} | Tenseur de rigidité |
| D^\pm | Opérateur de différence finie avant et arrière |
| D_{Ij} | Opérateur différentiel reliant les déplacements de la direction j à la contrainte I |
| f | Fréquence |
| $F(\cdot)$ | Transformée de Fourier |
| $g_{n,N}$ | Coefficient de l'opérateur de différence finie |
| $H(t)$ | Fonction de Heavyside |
| $J_{\alpha\beta}$ | Matrice jacobienne |
| k | Nombre d'onde |
| L | Nombre de mécanisme de relaxation |
| M | Module des ondes P |
| N_m | Nombre de paramètres d'inversion |
| N_x | Nombre de point de la grille en différence finie dans la direction x |
| N_{DF} | Demi-longueur du schéma en différence finie |
| Q | Facteur de qualité |
| R_I^l | Variable de mémoire modifiée du mécanisme de relaxation l |
| r_I^l | Variable de mémoire du mécanisme de relaxation l |
| s_{σ_I} | Terme source de la composante I de la contrainte |
| s_{v_i} | Terme source de la vitesse des particules dans la direction i |
| u_i | Déplacement des particules dans la direction i |
| v_i | Vitesse des particules dans la direction i |
| V_p | Vitesse des ondes de compression |
| V_s | Vitesse des ondes de cisaillement |

Chapitre 1

Introduction

Imager l'intérieur d'un solide est un défi technique rencontré dans une panoplie de domaines, et ce, à différentes échelles. Ainsi, le sismologue souhaite étudier la structure de la Terre, du noyau au manteau (Rawlinson *et al.*, 2010; Dahlen & Tromp, 1998). Les géophysiciens d'exploration veulent imager les réservoirs pétroliers situés à plusieurs kilomètres sous Terre (Yilmaz, 2001). L'hydrogéologue doit connaître la structure des aquifères pour comprendre l'écoulement des eaux et le transport des contaminants (Rubin & Hubbard, 2006). L'ingénieur civil doit vérifier l'intégrité des structures en béton (Malhotra & Carino, 2003). Le médecin a besoin de détecter la présence de tumeurs (Weissleder & Pittet, 2008; Brindle, 2008). Le biologiste désire étudier le fonctionnement des cellules *in vivo* (Pérez-Cota *et al.*, 2016). Quoique l'échelle de ces problématiques varie de l'ordre du nanomètre aux milliers de kilomètres, l'objectif demeure le même: caractériser la structure interne et mesurer les propriétés physiques d'un solide sans avoir un accès direct au médium, soit par l'impossibilité physique de l'atteindre (sismologie), par manque de ressources financières (hydrogéologie, exploration pétrolière), ou pour préserver son intégrité (génie civil, imagerie médicale).

Dans tous ces domaines, la propagation des ondes mécaniques est à la base de différentes technologies d'imagerie, qui se distinguent par leurs approximations physiques et la longueur d'onde des sources (des séismes d'une longueur d'onde de quelques centaines de mètres, à l'effet thermoélastique qui provoque des oscillations d'une longueur d'onde d'une fraction de micron). Une des méthodes les plus avancées en la matière a été développée dans le cadre de l'exploration pétrolière et de la sismologie; celle-ci se nomme l'inversion de formes d'ondes complètes. L'inversion de formes d'ondes complètes se distingue des autres méthodes d'imagerie en ce qu'elle tente de reproduire l'entièreté

des trains d’ondes enregistrés lors de l’auscultation du solide. Ceci est possible en modélisant l’équation d’onde complète dans un milieu hétérogène, en limitant le nombre d’approximations utilisées. Quoique développée pour des applications particulières et une échelle particulière, l’inversion de formes d’ondes complètes est une technologie générale qui peut en théorie être appliquée à plusieurs problématiques de différentes échelles.

Quelques écueils limitent tout de même la démocratisation de cette technologie, tant pour les différents domaines des sciences de la Terre qu’au sens large, en sciences appliquées. Un des principaux défis est sans aucun doute son coût exorbitant. En effet, en raison des puissances de calcul requises lors de la modélisation numérique de l’équation d’onde, l’inversion de formes d’ondes complètes requiert, pour des problèmes de tailles somme toute raisonnables, des ressources informatiques gargantuesques, propulsant l’industrie pétrolière au premier rang des consommateurs de calcul haute performance (Reuters, 2013; BP, 2013). De plus, l’acquisition et le traitement des levés sismiques pétroliers sont sujets à des contraintes particulières au domaine, faisant en sorte que l’inversion de formes d’ondes complètes ne commence à être une technologie mature que pour l’imagerie acoustique, négligeant plusieurs phénomènes ondulatoires d’intérêt en d’autres circonstances. Dans une visée globale d’élargir les champs d’application de cette technologie, particulièrement en sciences de la Terre, cette thèse aborde deux questions principales. Premièrement, comment réduire le coût numérique de l’inversion de formes d’ondes complètes, notamment en tirant avantage du niveau de parallélisme toujours croissant des ordinateurs modernes ? Deuxièmement, est-il possible de réduire le degré d’approximation physique de l’équation d’onde afin de reconstruire davantage de propriétés physiques avec l’inversion de formes d’ondes complètes ? En particulier, est-il possible d’inclure et d’inverser les phénomènes viscoélastiques, qui sont d’intérêts dans nombreuses applications ?

1.1 Vers une estimation quantitative des propriétés physiques

L’amélioration des technologies d’imagerie est motivée par le besoin d’une estimation quantitative plus précise des propriétés du sous-sol et des fluides qu’il contient. En effet, dans bien des domaines, les mesures des propriétés physiques des sols peuvent être corrélées avec des propriétés d’intérêt primaire. Par exemple, en géotechnique, la vitesse des ondes P est reliée au *Rock Quality Designation* (RQD) (Sjøgren *et al.*, 1979), un indice largement utilisé pour évaluer la capacité portante des roches (Bieniawski, 1989). En hydrogéologie, la résistivité et la permittivité électrique des

sols dépendent du contenu en eau et de la porosité Singh & Shah (2005). Les vitesses des ondes P sont aussi influencées par le contenu en fluide des roches, et particulièrement par la présence d'huile ou de gaz (Toksöz *et al.*, 1976), une des raisons pour lesquelles les levés sismiques sont autant employés en exploration pétrolière.

Il n'est cependant pas toujours possible d'utiliser directement ces relations pétrophysiques pour estimer une propriété d'intérêt. En effet, souvent, les méthodes d'interprétation et d'acquisition en géophysique ne permettent pas une reconstruction assez juste ou d'assez haute résolution pour un traitement quantitatif. Ainsi, dans bien des domaines, la géophysique est vue comme un outil de détection d'anomalies, pour lesquelles des investigations plus approfondies sont requises. Par exemple, les levés de sismique réflexion sont traditionnellement utilisés comme un outil de caractérisation lithologique devant être interprétés selon le contexte géologique estimé aux puits pour caractériser des réservoirs potentiels (Brown, 1996).

Une intégration quantitative des mesures géophysiques est cependant de plus en plus souhaitée. Cette intégration passe souvent par une approche géostatistique (Pyrzcz & Deutsch, 2014), dans laquelle les propriétés géophysiques sont traitées conjointement aux mesures pétrophysiques directes, c'est-à-dire mesurées aux puits. Par exemple, la modélisation de la porosité et autres propriétés économiques des réservoirs pétroliers bénéficie grandement de l'intégration des données sismiques à l'aide de relations pétrophysiques locales, tel que présenté par Eidsvik *et al.* (2004), Skjervheim *et al.* (2005) ou Bosch *et al.* (2010). Pour pouvoir être utilisées dans de telles approches, les propriétés estimées par la géophysique doivent être quantitatives et doivent avoir une assez haute résolution afin de pouvoir caractériser et inférer les propriétés géologiques et économiques du réservoir à l'échelle du réservoir. Ceci est le principal défi pour la géophysique et est la principale raison pourquoi des techniques comme l'inversion de formes d'ondes complètes sont mises de l'avant.

1.2 Méthodes sismiques

Évidemment, de nombreux phénomènes physiques peuvent être utilisés pour obtenir des informations quantitatives sur l'intérieur d'un solide et sur la sous-surface en particulier. Mentionnons notamment différents champs de potentiels (gravimétrique, électrique, magnétique), les ondes électromagnétiques, les particules hautes énergies (radioactivité), et bien évidemment les ondes méca-

niques. L'avantage des ondes mécaniques par rapport aux autres phénomènes physiques découle de leur capacité à se propager sur de grandes distances sans être trop atténuées, de leur pouvoir de résolution élevée, propre aux phénomènes ondulatoires, et de la diversité des échelles pour lesquelles des vibrations mécaniques peuvent être enregistrées et traitées, de l'échelle planétaire à l'échelle sous-micrométrique. De plus, les propriétés mécaniques pouvant être estimées par leur traitement sont d'intérêt dans une panoplie de disciplines, de la sismologie au génie civil. L'amélioration des techniques d'imagerie basée sur les ondes mécaniques a donc une portée vaste en science de la Terre, et même au-delà.

1.2.1 Les méthodes traditionnelles et leurs limites

En géophysique, les ondes mécaniques sont utilisées pour étudier la structure du sous-sol depuis la fin du 19e siècle (Aki & Richards, 2002). Notamment, les ondes produites par les séismes ont mené à la découverte du Moho par Andrija Mohorovicic (Jarchow & Thompson, 1989). Les levés sismiques employant des sources actives ont été développés peu après pour la prospection pétrolière, premièrement avec la méthode de la sismique réfraction, puis avec la sismique réflexion. Depuis, une panoplie de méthodes de traitement des ondes sismiques ont émergé, chacune basée sur un mode de propagation différent de l'énergie sismique, apportant des informations différentes sur le sous-sol.

Une des plus vieilles méthodes actives, la sismique réfraction, est basée sur l'analyse des premières arrivées des ondes sismiques (P ou S , selon le type de source). Dans la plupart des approches de traitement en réfraction (Hagedoorn, 1959; Hawkins, 1961; Palmer, 1981), le sous-sol est traité comme une série de couches homogènes dans lesquelles se propage un train d'onde de fréquence infinie (approximation de rais). L'approximation d'un modèle en couche peut-être assouplie en utilisant la tomographie par tracé de rais (Rawlinson *et al.*, 2010). Or, dans tous les cas, chaque trace sismique est réduite à une seule valeur numérique, le temps de pointé de la première arrivée, et l'aspect ondulatoire de la propagation des ondes est négligé.

La méthode sismique qui est sans conteste la plus utilisée à ce jour est la sismique réflexion. Les réflexions causées par des contrastes de propriétés mécaniques du sous-sol peuvent être traitées selon des méthodes plus ou moins complexes, de la sommation en point milieu commun (Yilmaz, 2001) à la migration à temps inverse (Baysal *et al.*, 1983). Au final, la sismique réflexion donne une image haute résolution de la réflectivité du sol, indiquant les contacts des changements d'impédance

acoustique. L'information est cependant semi-quantitative: l'estimation des vitesses, sous produit du traitement, est trop imprécise et de trop faible résolution pour être utilisée en pratique. Une estimation plus quantitative passe par des analyses plus poussées, comme l'analyse des amplitudes selon le déport (Ostrander, 1984; Rutherford & Williams, 1989), l'inversion colorée (Lancaster & Whitcombe, 2000), et bien d'autres (Russell & Dommico, 1988).

Un autre exemple de traitement sismique se focalisant sur un seul type d'arrivée est l'analyse multicanal des ondes de surface (MASW). Le caractère dispersif des ondes de surface est connu depuis les travaux de Lord Rayleigh (Strutt, 1885). La dispersion des ondes de surface, c'est-à-dire la dépendance de la vitesse de phase à la fréquence, est contrôlée principalement par la distribution en profondeur de la vitesse de cisaillement du solide. Ce n'est que bien plus tard, cependant, que la possibilité de retrouver les vitesses de cisaillement par le traitement des ondes de surface a été reconnue (Jones, 1958; Dorman & Ewing, 1962). Cette approche est devenue mature avec le MASW proposé par Park *et al.* (1999). Telle qu'utilisée actuellement, l'analyse des ondes de surface est très réductrice: elle suppose un modèle 1D en couches et réduit chaque tir sismique à une ou plusieurs courbes de dispersion du mode fondamental et des modes supérieurs des ondes de surface. La profondeur d'investigation des ondes de surface est aussi limitée par la fréquence la plus basse pouvant être générée par la source ou enregistrée au récepteur.

Il va sans dire que la liste des méthodes ici présentée n'est pas exhaustive. Ces méthodes, dites standards par opposition à l'inversion de formes d'ondes complètes, reposent cependant sur le même principe de réduction: réduction des données, en extrayant une infime partie de l'information contenue dans un seul type d'arrivée, et réduction de la physique, en approximant l'équation d'onde à un régime particulier. Cette réduction a comme conséquences une perte de résolution par rapport aux capacités théoriques des phénomènes ondulatoires, et la difficulté, voire l'impossibilité, de retrouver tous les paramètres mécaniques simultanément, de façon cohérente.

1.2.2 L'inversion de formes d'ondes complètes

Par contraste, l'inversion de formes d'ondes complètes vise à réduire le niveau d'approximation sur la physique de propagation des ondes à son strict minimum. Lailly (1983) puis Tarantola (1984a) ont montré qu'il est en effet possible de traiter les tirs sismiques en utilisant directement les équations différentielles partielles de l'équation d'onde pour retrouver les vitesses des ondes P ,

de même que toutes les propriétés viscoélastiques (Tarantola, 1988). En résolvant l'équation d'onde numériquement, il est possible de calculer l'amplitude et la phase de toutes les arrivées sismiques pour un milieu hétérogène quelconque. Ainsi, pour estimer les propriétés mécaniques d'un solide, il suffit de trouver un modèle qui reproduit adéquatement les mesures complètes des champs d'ondes enregistrés. Comme l'entièreté du train d'ondes est reproduite dans ce cas, l'inversion de formes d'ondes complètes évite les pertes d'informations inhérentes aux méthodes traditionnelles.

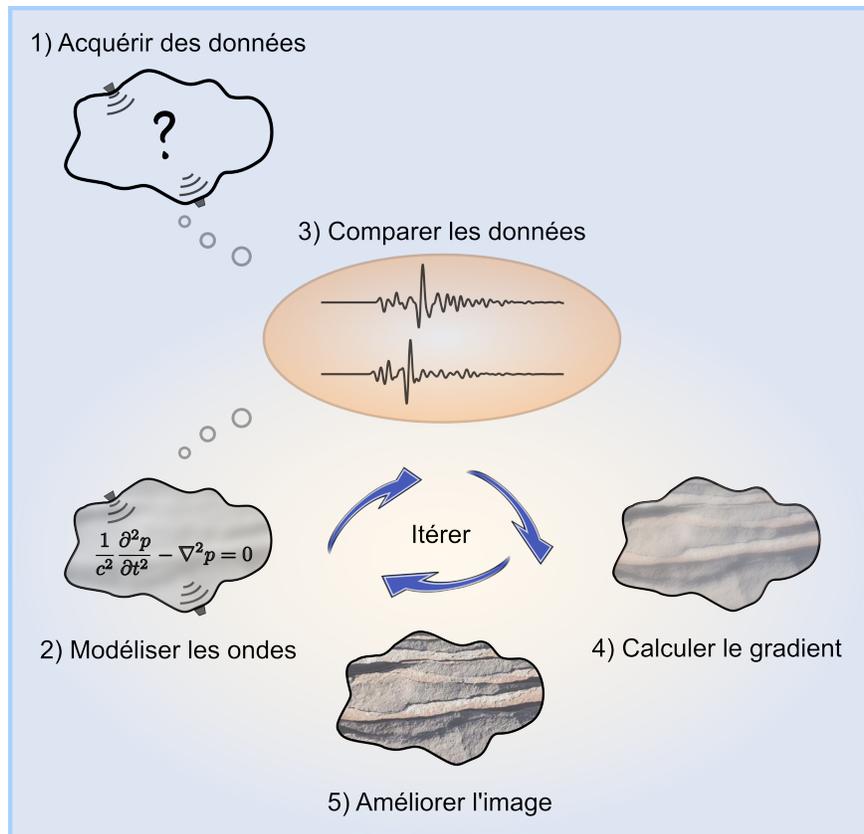


Figure 1.1 – Schéma de l'inversion de formes d'ondes complètes.

Le schéma de la Figure 1.1 illustre plus en détail les étapes de l'inversion de formes d'ondes complètes. Dans un premier temps, des enregistrements d'un champ d'onde se propageant à l'intérieur du solide à caractériser sont acquis. Le type d'enregistrement reste ici général, car les mesures peuvent être faites dans différents contextes (sismologie globale, sismique d'exploration ou même tests à ultrasons). L'inversion débute en posant un modèle initial des paramètres mécaniques, incluant généralement les informations *a priori* connues sur le solide étudié. À partir de ce modèle, l'équation d'onde est résolue numériquement afin de simuler le champ d'ondes qui aurait été acquis si ce modèle représentait la structure du solide. La troisième étape consiste à comparer directement les traces acquises et simulées. Comme, selon toute vraisemblance, le modèle initial ne correspondra

pas parfaitement à la structure réelle du sol, les données modélisées ne reproduiront pas exactement les données enregistrées, entraînant ainsi une différence, appelée résidu. Ce résidu permet de calculer une correction à apporter au modèle selon la méthode adjointe (Chavent, 1974). Cette correction, c'est-à-dire le gradient de la fonction coût, est utilisée pour améliorer le modèle initial selon une stratégie itérative d'optimisation locale. Ce processus de prédiction et correction est répété jusqu'à ce que l'accord entre les traces modélisées et enregistrées soit satisfaisant.

1.3 Problématique

Depuis les premières formulations de Tarantola (1984a, 1988), l'inversion de formes d'ondes complètes est devenue un sujet d'intenses recherches, surtout avec l'avènement de nouvelles méthodes d'acquisition permettant de plus longs espacements entre les sources et les récepteurs (déports) et contenant des basses fréquences. Deux formulations modernes sont largement utilisées: la formulation dans le temps de Mora (1987) et la formulation en fréquence de Pratt (1990). Quoique la théorie a été développée pour l'équation anisotrope viscoélastique dès le départ par Tarantola (1988), ces deux approches sont la plupart du temps appliquées dans le régime acoustique isotrope: Operto *et al.* (2006); Bleibinhaus *et al.* (2007) appliquent l'inversion de formes d'ondes complètes à un jeu de données réels 2D, (Ben-Hadj-Ali *et al.*, 2008; Vigh *et al.*, 2011; Baeten *et al.*, 2013; Schiemenz & Igel, 2013) présentent des études de cas en 3D. Le régime élastique est plus rarement étudié, en raison de son coût de calcul beaucoup plus important, mais aussi en raison de la non-linéarité plus importante causée par le plus grand nombre de paramètres à inverser. Des études de cas existent tout de même, tant en 2D (Shipp & Singh, 2002; Brossier *et al.*, 2009) qu'en 3D (Guasch *et al.*, 2012).

Le régime viscoélastique n'a été que très peu abordé dans la littérature, étant donné son coût encore plus élevé que le régime élastique. Néanmoins, l'atténuation visqueuse peut avoir un impact significatif sur la propagation des ondes, particulièrement dans des dépôts non-consolidés (Prasad *et al.*, 2004; Zimmer *et al.*, 2007). De plus, de nombreuses mesures en laboratoire et sur le terrain montrent que le facteur de qualité Q , soit une mesure de la magnitude de l'atténuation, est corrélé à de nombreuses propriétés d'intérêt en sciences de la Terre (Barton, 2007), comme le type de roche et la saturation en fluide. Ainsi, le facteur de qualité est une propriété physique d'intérêt en soi, tant pour les ondes P que pour les ondes S . Pour élargir les champs d'application de l'inversion

de formes d’ondes complètes et obtenir des estimations justes des propriétés mécaniques des solides, particulièrement lors de levés sismiques de proche surface dans des milieux non-consolidés, la viscoélasticité devrait être considérée lors de l’inversion.

De nombreuses approches théoriques ont été proposées pour l’inversion de formes d’ondes complètes viscoélastique, la plupart dans le domaine fréquentiel (Song *et al.*, 1995) ou dans le domaine de Laplace-Fourier (Kamei & Pratt, 2013). La simplicité d’inclure l’atténuation dans le domaine fréquentiel est l’un de ses principaux avantages par rapport à l’approche dans le domaine temporel: il suffit de définir des vitesses complexes pour implémenter un profil d’atténuation donné en fréquence (Toksöz *et al.*, 1976). À l’opposé, l’approche dans le domaine temporel nécessite de résoudre des équations différentielles partielles additionnelles pour des variables de mémoire (Carcione *et al.*, 1988a; Robertsson *et al.*, 1994), et l’obtention d’un profil d’atténuation en fréquence d’une forme donnée n’est pas simple (Blanch *et al.*, 1995). Cependant, l’approche dans le temps par différences finies demeure particulièrement utile pour des modèles 3D de grande taille, en raison de sa capacité inhérente à être parallélisée (Fichtner, 2011). En profitant de ce parallélisme sur des plateformes de calcul modernes, il serait possible de réduire à un niveau acceptable le coût numérique de l’inversion viscoélastique.

La littérature sur l’inversion viscoélastique dans le domaine du temps est beaucoup plus ténue que dans le domaine des fréquences. La plupart des auteurs ne considèrent en fait que le cas viscoacoustique (Liao & McMechan, 1995, 1996; Causse *et al.*, 1999; Bai *et al.*, 2014). Lorsque la viscoélasticité est considérée, comme dans les travaux de Tarantola (1988) ou plus récemment de Charara *et al.* (2000), Fichtner *et al.* (2006) et Askan *et al.* (2007), la formulation du déplacement de l’équation d’onde est utilisée. Cependant, ce formalisme n’est pas directement applicable à la formulation vitesses-contrainte popularisée par Robertsson *et al.* (1994) pour les milieux viscoélastiques, étant donné que les équations adjointes ne sont pas les mêmes pour ces deux formulations (Castellanos *et al.*, 2011). En bref, une implémentation pratique de l’inversion de formes d’ondes complètes viscoélastique adaptée aux approches de modélisation dans le domaine temporel nécessite le développement de nouvelles équations adjointes.

1.4 Objectifs de la thèse

Dans cette thèse, nous nous pencherons plus en détail sur l'inversion de formes d'ondes complètes viscoélastique dans le domaine du temps. Afin de développer une approche d'inversion viscoélastique réaliste, trois objectifs sont formulés.

Le premier objectif est d'appliquer la méthode adjointe à la formulation vitesse-contrainte de l'équation d'onde viscoélastique. Ceci est l'objet du Chapitre 3. En particulier, à l'aide de transformations linéaires, des équations adjointes symétriques aux équations directes sont obtenues, permettant d'utiliser le même code numérique pour obtenir la solution des deux systèmes. Cette contribution originale est présentée dans un article soumis à *Geophysics*:

- Fabien-Ouellet, G., Gloaguen, E. & Giroux, B., (2016) Anisotropic velocity-stress full waveform inversion. Soumis à *Geophysics* en date du 30/11/2016.

Le deuxième objectif est de montrer comment une implémentation numérique utilisant le parallélisme des plateformes de calcul modernes permet de réduire le coût de calcul de l'inversion viscoélastique à un niveau raisonnable. Cette question sera abordée au Chapitre 4, dans lequel un code de résolution numérique des équations viscoélastiques directes et adjointes du chapitre 3 est développé à l'aide de l'interface de programmation OpenCL. Ce code utilise plusieurs niveaux de parallélismes, ce qui permet d'accélérer significativement les calculs lors de l'inversion. Ces résultats sont publiés dans:

- Fabien-Ouellet, G., Gloaguen, E., & Giroux, B. (2016). Time-domain seismic modeling in viscoelastic media for full waveform inversion on heterogeneous computing platforms with OpenCL. *Computers & Geosciences*. doi:10.1016/j.cageo.2016.12.004

Le dernier objectif est de déterminer s'il est possible avec l'inversion viscoélastique de reconstruire les vitesses des ondes P et des ondes S , ainsi que leur facteur de qualité, Q . Ceci est abordé au Chapitre 5, présentant une étude de cas synthétique appliquant l'inversion de formes d'ondes complètes viscoélastique au suivi temporel de l'injection de CO_2 . Les limitations de l'inversion des facteurs de qualité sont aussi abordées. Ces travaux font l'objet d'un article accepté avec révisions modérées:

- Fabien-Ouellet, G., Gloaguen, E., & Giroux, B. (2016). Time-domain viscoelastic full waveform inversion. Publication acceptée par *Geophysical Journal International* avec révisions modérées en date du 10/10/2016

La théorie nécessaire pour comprendre et mettre en perspective les résultats présentés dans cette thèse est abordée en premier au Chapitre 2.

Chapitre 2

Survol théorique

Le procédé itératif de reconstruction des paramètres physiques du sous-sol décrit à la Figure 1.1 n'est pas unique à l'inversion de formes d'ondes complètes. C'est en fait le domaine de ce que l'on nomme inversion en géophysique (Meju, 1994; Menke, 2012; Aster *et al.*, 2013). Typiquement, une série de mesures \mathbf{d} sont effectuées sur le médium à caractériser. Ces mesures peuvent être un potentiel électrique, le champ magnétique, une onde électromagnétique, et bien sur une onde mécanique. Celles-ci sont prédites par une théorie physique, ou ce que l'on nomme communément le modèle direct $\mathcal{F}(\mathbf{m})$. Par exemple, une onde électromagnétique peut être décrite par les équations de Maxwell, qui dépendent de paramètres physiques \mathbf{m} , dans ce cas, la permittivité et la conductivité électrique du médium. La prédiction des paramètres par les données mesurées, soit le contraire du modèle direct, se nomme l'inversion. Schématiquement:

- **Problème direct:** $\mathbf{d} = \mathcal{F}(\mathbf{m})$
- **Problème inverse:** $\mathbf{m} = \mathcal{F}^{-1}(\mathbf{d})$

En général, l'expression mathématique du problème inverse $\mathcal{F}^{-1}(\cdot)$ n'existe pas, le problème direct étant souvent non linéaire et mal posé. Le problème inverse est plutôt résolu en définissant une mesure de la discordance entre les données mesurées et les données prédites par le modèle direct, soit une fonction coût $\mathcal{X}(\mathbf{m}; \mathbf{d})$, qui doit être minimisée par un algorithme d'optimisation. Lorsque les données prédites concordent aux données mesurées, c'est-à-dire lorsque la fonction coût est minimum, le problème inverse est résolu.

Ce schéma général comprend trois éléments essentiels: une théorie physique, une approche de résolution numérique des équations directes et un algorithme d'optimisation. Dans le cas de l'inversion de formes d'ondes complètes, la théorie physique adoptée dans ce travail est l'équation d'onde viscoélastique, l'approche de résolution numérique est la méthode des différences finies et l'optimisation se fera par une méthode de descente locale. Ce chapitre aborde ces trois thèmes tour à tour, afin de construire la base théorique mettant en contexte les développements présentés aux chapitres subséquents.

2.1 Propagation des ondes

2.1.1 Élasticité linéaire

La propagation des ondes dans le sous-sol est souvent modélisée par la théorie de l'élasticité linéaire (voir Love (1892) pour une référence historique intéressante et Aki & Richards (2002) pour un traitement moderne). Selon celle-ci, un matériau solide se comporte comme un ressort pour de faibles contraintes σ , c'est-à-dire que la déformation ϵ est proportionnelle à la force appliquée et revient toujours à son état de repos lorsque les contraintes sont enlevées. Ceci correspond au régime élastique de la Figure 2.1.

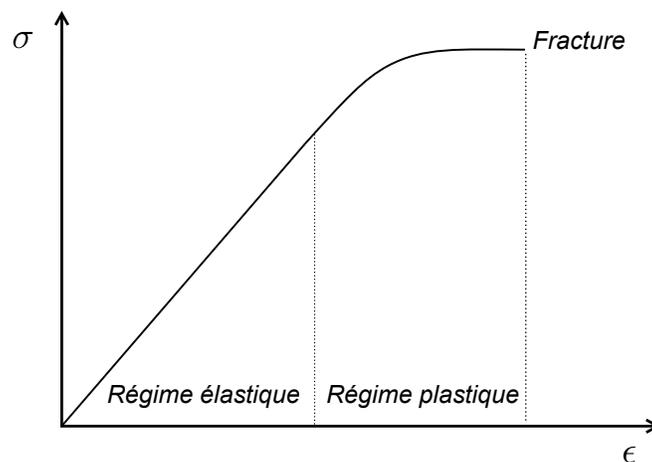


Figure 2.1 – Diagramme de contrainte déformation d'un solide, montrant le régime élastique valable pour de faibles contraintes.

Il existe ainsi une constante de proportionnalité entre les déformations et les contraintes. Comme une contrainte peut être appliquée selon plusieurs directions, en compression/traction ou en cisaille-

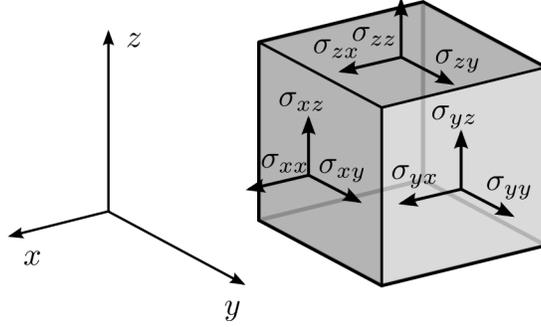


Figure 2.2 – Directions et plans des contraintes appliquées à un volume élémentaire d'un solide.

ment, la contrainte est un tenseur 3x3 à neuf composantes. Ces directions sont illustrées à la Figure 2.2. En raison de la conservation du moment angulaire, ce tenseur est symétrique et seulement 6 composantes sont indépendantes. Ceci permet d'écrire les contraintes sous la forme d'un vecteur à 6 composantes:

$$\boldsymbol{\sigma} = (\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{yz}, \sigma_{xz}, \sigma_{xy})^T, \quad (2.1)$$

ce qui correspond à la notation de Voigt (Voigt, 2014). De même, les déformations peuvent s'écrire

$$\boldsymbol{\epsilon} = (\epsilon_{xx}, \epsilon_{yy}, \epsilon_{zz}, 2\epsilon_{yz}, 2\epsilon_{xz}, 2\epsilon_{xy})^T. \quad (2.2)$$

Les constantes de proportionnalité entre contraintes et déformations sont données par le tenseur de rigidité c_{ijkl} . Ce tenseur d'ordre 4 comporte $3^4 = 81$ composantes, dont seulement 21 sont indépendantes (Green, 1871). Ainsi, selon la notation de Voigt, le tenseur de rigidité peut être exprimé par la matrice suivante:

$$\boldsymbol{\Psi}^e = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & c_{15} & c_{16} \\ & c_{22} & c_{23} & c_{24} & c_{25} & c_{26} \\ & & c_{33} & c_{34} & c_{35} & c_{36} \\ & & & c_{44} & c_{45} & c_{46} \\ & & & & c_{55} & c_{56} \\ & & & & & c_{66} \end{bmatrix}. \quad (2.3)$$

La relation contraintes-déformations dans le régime linéaire élastique peut être exprimée comme:

$$\sigma_I = \Psi_{IJ}^e \epsilon_J. \quad (2.4)$$

2.1.2 Viscoélasticité linéaire

La théorie de l'élasticité linéaire néglige cependant un facteur important: la dissipation de l'énergie mécanique, majoritairement sous forme de chaleur. La théorie de la viscoélasticité linéaire permet d'inclure ce terme dissipatif dans la relation contraintes-déformations. Christensen (2012) et Meyers & Chawla (2009) sont de bonnes références sur le sujet. La viscoélasticité linéaire assume une relation linéaire entre les contraintes et le taux de déformation. Ceci peut s'exprimer comme:

$$\sigma(t) = \int_{-\infty}^t \Psi(t-t') \partial_t \epsilon(t') dt'. \quad (2.5)$$

En d'autres termes, la contrainte est donnée par la convolution dans le temps entre la matrice de relaxation Ψ et le taux de déformation, soit

$$\sigma_I = \Psi_{IJ} * \partial_t \epsilon_J. \quad (2.6)$$

Le diagramme de contraintes-déformations viscoélastique diffère du régime élastique par l'hystérésis de la courbe qui est manifeste des pertes d'énergie mécanique (voir Figure 2.3).

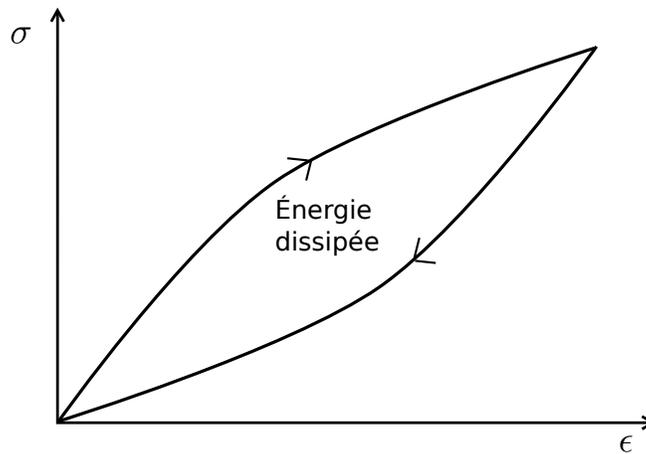


Figure 2.3 – Diagramme de contrainte déformation d'un solide viscoélastique, montrant que les contraintes dépendent de l'historique de déformation.

Plusieurs modèles rhéologiques existent pour décrire la matrice de relaxation, qui changeront le comportement dans le temps de la réponse en contrainte du solide. En sismique, notons notamment l'utilisation du modèle en loi de puissance (Strick, 1967; Kjartansson, 1979), le modèle de Maxwell et le modèle de Kelvin-Voigt (Casula & Carcione, 1992), le solide linéaire standard (Ben-Menahem & Singh, 2012) et le solide de Maxwell généralisé (SMG) (Emmerich & Korn, 1987; Carcione *et al.*,

1988a,b; Moczo, 2005). De loin le plus utilisé est le SMG, qui est une superposition en parallèle de L modèles de Maxwell, c'est-à-dire un ressort de constante k_l connecté en série avec un amortisseur de viscosité η_l . Ce modèle est illustré à la Figure 2.4. Selon Moczo (2005), le module d'élasticité du SMG est donné par:

$$M = M_e \left[1 - \frac{1}{L} \sum_{l=1}^L \left(1 - \frac{\tau_{el}}{\tau_{\sigma l}} \right) e^{-t/\tau_{\sigma l}} \right] H(t), \quad (2.7)$$

où τ_{el} et $\tau_{\sigma l}$ sont les temps de relaxation des déformations et des contraintes respectivement, et M_e est le module relaxé du solide.

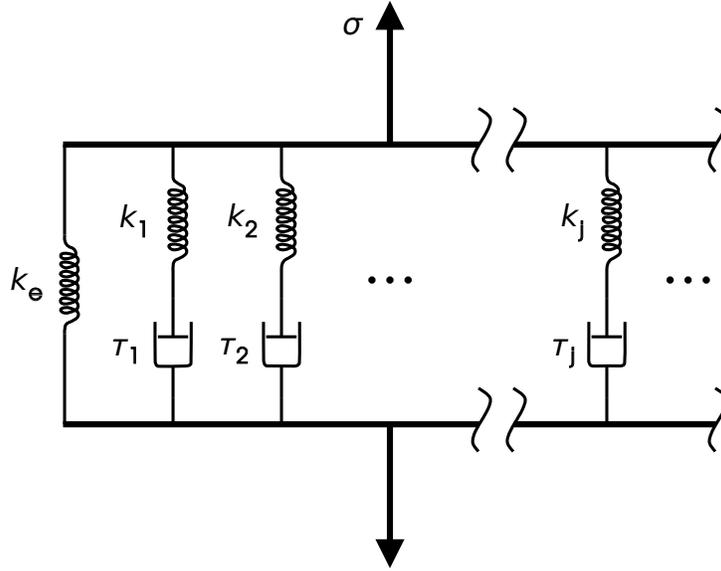


Figure 2.4 – Modèle du solide de Maxwell généralisé, consistant en une superposition en parallèle de plusieurs solides de Maxwell, constitués d'un ressort et un amortisseur en série.

L'atténuation d'une onde est souvent caractérisée par le facteur de qualité Q défini comme

$$Q = \frac{\Re(M(\omega))}{\Im(M(\omega))}, \quad (2.8)$$

où $\Re(\cdot)$ est la partie réelle et $\Im(\cdot)$ est la partie imaginaire. L'équation (2.7) peut être exprimée dans le domaine fréquentiel,

$$M(\omega) = \frac{M_e}{L} \sum_{l=1}^L \frac{1 + i\omega\tau_{el}}{1 + i\omega\tau_{\sigma l}}. \quad (2.9)$$

Ces deux équations permettent d'obtenir le facteur de qualité du SMG, soit

$$Q(\omega) = \frac{1 + \sum_{l=1}^L \tau_l \frac{\omega^2 \tau_{\sigma l}^2}{1 + \omega^2 \tau_{\sigma l}^2}}{\sum_{l=1}^L \tau_l \frac{\omega \tau_{\sigma l}}{1 + \omega^2 \tau_{\sigma l}^2}}, \quad (2.10)$$

où le niveau d'atténuation τ introduit par Blanch *et al.* (1995) est défini comme

$$\tau = 1 - \frac{\tau_{el}}{\tau_{\sigma l}}. \quad (2.11)$$

La courbe du facteur de qualité en fréquence pour un seul mécanisme d'atténuation est celle d'une cloche dont le centre en fréquence est contrôlé par le temps de relaxation $\tau_{\sigma l}$ et l'amplitude est contrôlée par le niveau d'atténuation τ . L'intérêt du solide de Maxwell généralisé est qu'il permet d'obtenir un profil d'atténuation en fréquence arbitraire en utilisant une superposition de mécanismes d'atténuation. En optimisant les temps de relaxation de plusieurs mécanismes (Blanch *et al.*, 1995; Bohlen, 2002), il est possible d'obtenir un profil plus ou moins constant en fréquence dont le niveau sera contrôlé par τ , comme illustré à la Figure 2.5. À noter qu'un profil d'atténuation constant sur une large bande de fréquences est préconisé en sismique (Liu *et al.*, 1976; Barton, 2007).

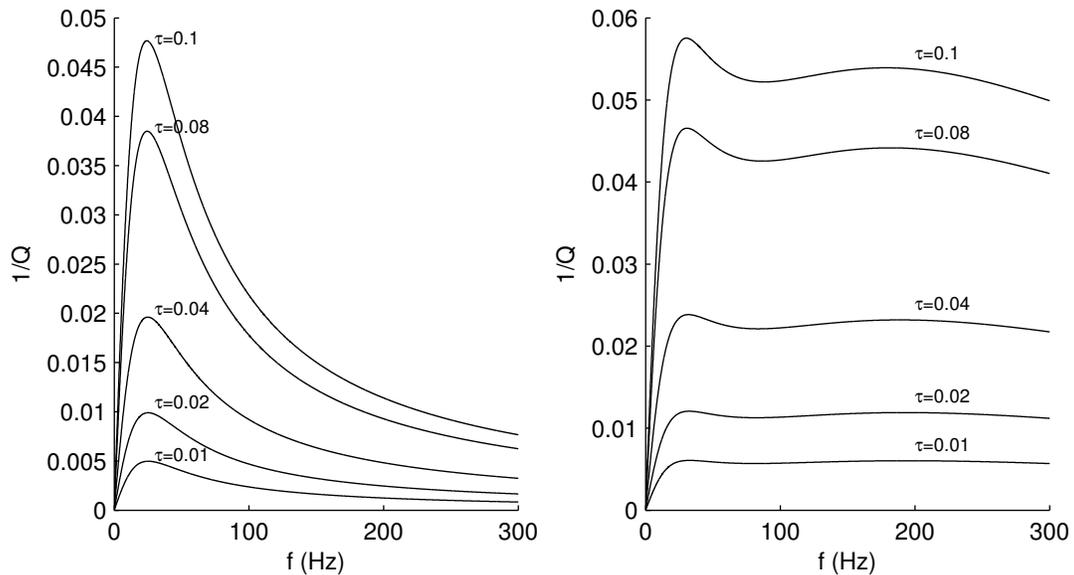


Figure 2.5 – Profil d'atténuation en fréquence du solide de Maxwell généralisé pour plusieurs niveaux d'atténuation, avec un seul mécanisme (droite) et deux mécanismes (gauche).

En généralisant ce modèle à toutes les composantes de la matrice de relaxation, nous pouvons écrire:

$$\Psi = (\Psi^e + \Psi^v \sum_{l=1}^L e^{-t/\tau_{\sigma l}})H(t), \quad (2.12)$$

avec l'opérateur différentiel \mathbf{D} défini comme

$$\mathbf{D} = \begin{bmatrix} \partial_x & 0 & 0 \\ 0 & \partial_y & 0 \\ 0 & 0 & \partial_z \\ 0 & \partial_z & \partial_y \\ \partial_z & 0 & \partial_x \\ \partial_y & \partial_x & 0 \end{bmatrix}. \quad (2.17)$$

Avec ces définitions, il est possible de réécrire l'équation (2.6) comme:

$$\sigma_I = \Psi_{IJ} * \partial_t \epsilon_J = \Psi_{IJ} * D_{Jk} v_k. \quad (2.18)$$

En remplaçant la matrice de relaxation donnée à l'équation (2.12), l'équation précédente devient

$$\sigma_I = (\Psi_{IJ}^e + \Psi_{IJ}^v \sum_{l=1}^L e^{-t/\tau_{\sigma l}}) H(t) * D_{Jk} v_k. \quad (2.19)$$

La dérivé par rapport au temps de l'équation précédente donne

$$\partial_t \sigma_I = (\Psi_{IJ}^e + \Psi_{IJ}^v) D_{Jk} v_k + \sum_{l=1}^L r_I^l, \quad (2.20)$$

où le vecteur des variables de mémoire \mathbf{r}^l est défini comme:

$$r_I^l = -\Psi_{IJ}^v \frac{1}{\tau_{\sigma l}} e^{-t/\tau_{\sigma l}} H(t) * D_{Jk} v_k. \quad (2.21)$$

Cette dernière équation est embêtante à résoudre numériquement en raison de l'opération de convolution. Cependant, il est possible d'obtenir une équation différentielle des variables de mémoire en prenant la dérivée dans le temps de l'équation précédente, soit

$$\partial_t r_I^l = -\frac{1}{\tau_{\sigma l}} \Psi_{IJ}^v D_{Jk} v_k - \frac{1}{\tau_{\sigma l}} r_I^l. \quad (2.22)$$

Enfin, la dernière équation requise pour obtenir l'équation d'onde est la deuxième loi de Newton,

$$\partial_t v_i - \frac{1}{\rho} D_{Ji} \sigma_J - s_{v_i} = 0, \quad (2.23)$$

où ρ est la densité et s_{v_i} est la force externe appliquée au solide. Les équations (2.20), (2.22) and (2.23) mises ensemble forment un système d'équations différentielles hyperboliques couplées, c'est-à-dire l'équation d'onde pour un milieu viscoélastique anisotrope dans la formulation vitesse-contrainte:

$$\begin{aligned} \partial_t v_i - \frac{1}{\rho} D_{Ji} \sigma_J - s_{v_i} &= 0, \\ \partial_t \sigma_I - \Psi_{IJ}^0 D_{Jk} v_k - \sum_{l=1}^{l=L} r_I^l - s_{\sigma_I} &= 0, \\ \partial_t r_I^l + \frac{1}{\tau_{\sigma l}} \Psi_{IJ}^l D_{Jk} v_k + \frac{1}{\tau_{\sigma l}} r_I^l &= 0, \end{aligned} \tag{2.24}$$

avec $\Psi^0 = \Psi^e + \Psi^v$ et $\Psi^{l>0} = \Psi^v$. Cette dernière équation est l'équation principale qui sera utilisée dans ce travail.

2.1.4 Formulation du déplacement

Notons cependant que la théorie de l'inversion de formes d'ondes complètes a, au départ, été développée par Tarantola (1984a) avec la formulation du déplacement. Celle-ci est donnée par:

$$\begin{aligned} \partial_{tt} u_i - \frac{1}{\rho} D_{Ji} \left[\Psi_{JK}^0 D_{Kk} u_k + \sum_{l=1}^{l=L} R_J^l \right] - s_{u_i} &= 0, \\ \partial_t R_I^l + \frac{1}{\tau_{\sigma l}} \Psi_{IJ}^l D_{Jk} u_k + \frac{1}{\tau_{\sigma l}} R_I^l &= 0, \end{aligned} \tag{2.25}$$

où $\partial_t R_J^l = r_J^l$. La formulation vitesse-contrainte a été choisie dans ce travail en raison de sa grande popularité pour la modélisation de l'équation d'onde viscoélastique dans le domaine du temps (Robertsson *et al.*, 1994; Bohlen, 2002), et en raison du calcul explicite des contraintes, qui est utile dans le cas de l'inversion en ondes complète avec des sources explosives et des hydrophones. Notons cependant que la résolution numérique est tout aussi possible pour la formulation en déplacement, notamment avec l'approche de Newmark (Newmark, 1959; Carcione, 1990)

2.1.5 Formulation en fréquence

Une des approches les plus utilisées en inversion de formes d'ondes complètes est l'approche en fréquence (Pratt, 1999; Sirgue & Pratt, 2004). Pour celle-ci, notons que la convolution de la relation

contrainte déformation donnée à l'équation (2.18) devient une simple multiplication dans le domaine des fréquences. En combinant cette équation avec l'équation (2.23) et en prenant la transformée de Fourier, l'équation d'onde en fréquence est obtenue:

$$\left(\omega^2 \rho - D_{Ji} \Psi_{JK}^{Im} D_{Kk}\right) \hat{u}_k(\omega) = \mathbf{F} \hat{\mathbf{u}}(\omega) = \hat{\mathbf{s}}(\omega). \quad (2.26)$$

Lorsque les dérivées dans l'espace sont discrétisées par différences finies, la matrice \mathbf{F} peut être inversée directement, par exemple par décomposition LU, pour donner la solution de l'équation à la fréquence ω . Une fois que l'inverse de la matrice \mathbf{F} est calculé pour une fréquence, il est possible d'obtenir directement la solution pour n'importe quelle position de la source à faible coût. Ceci représente le plus grand avantage de l'approche fréquentielle. Cependant, pour des problèmes 3D et élastique, la taille de cette matrice devient prohibitive (Gosselin-Cliche & Giroux, 2014), et l'approche en fréquence devient moins attrayante avec la technologie actuelle.

2.1.6 Approximation isotrope

Une approximation courante est l'approximation isotrope, c'est-à-dire que les propriétés visco-élastiques du milieu ne dépendent pas de la direction. Dans ce cas, $c_{11} = c_{22} = c_{33} = \lambda + 2\mu$, $c_{44} = c_{55} = c_{66} = \mu$, $c_{12} = c_{13} = c_{23} = \lambda$ et le reste des éléments de la matrice de rigidité sont nuls. Le nombre de paramètres passe ainsi de 21 composantes à uniquement 2, où λ est la première constante de Lamé et μ est le module de cisaillement. Dans ce cas, la matrice de relaxation de l'équation (2.12) peut s'écrire comme

$$\Psi_{ijkm}^l = \lambda^l \delta_{ij} \delta_{km} + \mu^l (\delta_{ik} \delta_{jm} + \delta_{im} \delta_{jk}), \quad (2.27)$$

avec

$$\lambda^l = \begin{cases} M(1 + \tau_p) - 2\mu(1 + \tau_s) & \text{if } l = 0, \\ M\tau_p - 2\mu\tau_s & \text{if } l > 0, \end{cases} \quad (2.28a)$$

$$\mu^l = \begin{cases} \mu(1 + \tau_s) & \text{if } l = 0, \\ \mu\tau_s & \text{if } l > 0. \end{cases} \quad (2.28b)$$

Le module des ondes P est donné par $M = \lambda + 2\mu$. L'équation d'onde (2.24) devient ainsi

$$\partial_t v_i - \frac{1}{\rho} \partial_j \sigma_{ij} - s_{v_i} = 0, \quad (2.29a)$$

$$\partial_t \sigma_{ij} - \lambda^0 \partial_k v_k \delta_{ij} - \mu^0 (\partial_j v_i + \partial_i v_j) - \sum_{l=1}^{l=L} r_{ij}^l - s_{\sigma_{ij}} = 0, \quad (2.29b)$$

$$\partial_t r_{ij}^l + \frac{1}{\tau_{\sigma l}} [\lambda^l \partial_k v_k \delta_{ij} + \mu^l (\partial_j v_i + \partial_i v_j) + r_{ij}^l] = 0. \quad (2.29c)$$

2.1.7 Approximation élastique

L'approximation élastique est obtenue lorsque τ_p et τ_s tendent vers 0. Ainsi, l'équation (2.24) se réduit à

$$\begin{aligned} \partial_t v_i - \frac{1}{\rho} D_{Ji} \sigma_J - s_{v_i} &= 0, \\ \partial_t \sigma_I - \Psi_{IJ}^0 D_{Jk} v_k - s_{\sigma_I} &= 0. \end{aligned} \quad (2.30)$$

Plus particulièrement, dans le cas isotrope:

$$\partial_t v_i - \frac{1}{\rho} \partial_j \sigma_{ij} - s_{v_i} = 0, \quad (2.31a)$$

$$\partial_t \sigma_{ij} - \lambda \partial_k v_k \delta_{ij} - \mu (\partial_j v_i + \partial_i v_j) - s_{\sigma_{ij}} = 0. \quad (2.31b)$$

Ainsi, les variables de mémoire sont éliminées, ce qui réduit la complexité de la résolution numérique de l'équation d'onde.

2.1.8 Approximation 2D

Souvent, les levés sismiques sont acquis le long d'un profil qui est perpendiculaire à la direction structurale de la géologie à caractériser. Dans ce cas, il peut être avantageux de réduire la dimensionnalité de l'équation d'onde de 3 à 2. Pour ce faire, deux régimes sont intéressants: le régime P - SV pour lequel le terme source est contenu dans le même plan que la propagation des ondes, et le régime SH , pour lequel la source est perpendiculaire au plan de propagation. L'équation d'onde P - SV peut être obtenue en considérant comme nulles les dérivées spatiales par rapport à une direction (en y par exemple). Dans ce cas, le plan xz doit être coplanaire à la direction de la source.

Ceci donne explicitement, pour l'équation d'onde élastique isotrope,

$$\partial_t v_x - \frac{1}{\rho} \partial_x \sigma_{xx} - \frac{1}{\rho} \partial_z \sigma_{xz} - s_{v_x} = 0, \quad (2.32a)$$

$$\partial_t v_z - \frac{1}{\rho} \partial_z \sigma_{zz} - \frac{1}{\rho} \partial_x \sigma_{xz} - s_{v_z} = 0, \quad (2.32b)$$

$$\partial_t \sigma_{xx} - (\lambda - 2\mu) \partial_x v_x - \lambda \partial_z v_z - s_{\sigma_{xx}} = 0, \quad (2.32c)$$

$$\partial_t \sigma_{zz} - (\lambda - 2\mu) \partial_z v_z - \lambda \partial_x v_x - s_{\sigma_{zz}} = 0, \quad (2.32d)$$

$$\partial_t \sigma_{xz} - \mu (\partial_x v_z + \partial_z v_x) - s_{\sigma_{xz}} = 0. \quad (2.32e)$$

L'équation d'onde *SH* est obtenue en considérant les composantes de la vitesse dans deux directions comme nulles (en x et z par exemple). Dans ce cas-ci, la source doit être dans une direction perpendiculaire au plan xz , soit en y , et nous avons

$$\partial_t v_y - \frac{1}{\rho} \partial_x \sigma_{yx} - \frac{1}{\rho} \partial_z \sigma_{yz} - s_{v_y} = 0, \quad (2.33a)$$

$$\partial_t \sigma_{yz} - \mu \partial_z v_y - s_{\sigma_{yz}} = 0, \quad (2.33b)$$

$$\partial_t \sigma_{yx} - \mu \partial_x v_y - s_{\sigma_{yx}} = 0. \quad (2.33c)$$

2.1.9 Solution analytique

La solution analytique du problème 3D isotrope viscoélastique pour un solide homogène et infini est bien connue (Aki & Richards, 2002). Pilant (2012) donne une fonction explicite de cette solution pour une force en z , qui est donnée par (en suivant Gosselin-Cliche & Giroux (2014)):

$$u_x = \frac{f(\omega)}{4\pi\rho|\mathbf{x}|^5\omega^2} \left[xz \left(|\mathbf{x}|^2 k_p^2 - 3 - 3i|\mathbf{x}|k_p \right) e^{-ik_p|\mathbf{x}|} - \left(|\mathbf{x}|^2 k_s^2 - 3 - 3i|\mathbf{x}|k_s \right) e^{-ik_s|\mathbf{x}|} \right] \quad (2.34a)$$

$$u_y = \frac{f(\omega)}{4\pi\rho|\mathbf{x}|^5\omega^2} \left[yz \left(|\mathbf{x}|^2 k_p^2 - 3 - 3i|\mathbf{x}|k_p \right) e^{-ik_p|\mathbf{x}|} - \left(|\mathbf{x}|^2 k_s^2 - 3 - 3i|\mathbf{x}|k_s \right) e^{-ik_s|\mathbf{x}|} \right] \quad (2.34b)$$

$$\begin{aligned} u_z = \frac{f(\omega)}{4\pi\rho|\mathbf{x}|^5\omega^2} & \left[\left(x^2 + y^2 - 2z^2 \right) \left(e^{-ik_p|\mathbf{x}|} - e^{-ik_s|\mathbf{x}|} \right) \right. \\ & + \left(z^2 |\mathbf{x}|^2 k_p^2 + i \left(x^2 + y^2 - 2z^2 \right) |\mathbf{x}| k_p \right) e^{-ik_p|\mathbf{x}|} \\ & \left. + \left((x^2 + y^2) |\mathbf{x}|^2 k_s^2 - i \left(x^2 + y^2 - 2z^2 \right) |\mathbf{x}| k_s \right) e^{-ik_s|\mathbf{x}|} \right], \end{aligned} \quad (2.34c)$$

où $k_p = \omega \sqrt{\frac{\rho}{M}}$ et $k_s = \omega \sqrt{\frac{\rho}{\mu}}$. Il existe aussi des solutions analytiques pour un milieu homogène semi-infini (Lamb, 1904; Miller & Pursey, 1954; Cherry, 1962) et un milieu en couches (Bouchon *et al.*, 1989). Aucune solution n'existe toutefois pour une distribution hétérogène des propriétés viscoélastiques du solide.

2.2 Résolution par différences finies de l'équation d'onde

2.2.1 Choix de la méthode

Dans le cas d'un solide ayant une distribution spatiale des paramètres viscoélastiques hétérogène, l'équation d'onde doit être résolue à l'aide de méthodes numériques, qui consistent à discrétiser les fonctions continues présentées aux sections précédentes. Lapidus & Pinder (2011) est une bonne référence sur la résolution numérique d'équations différentielles. On peut aussi consulter Virieux *et al.* (2011) pour une revue des méthodes de résolution numérique utilisées dans le domaine de l'inversion de formes d'ondes complètes. La liste suivante décrit sommairement les méthodes les plus utilisées dans le domaine.

- **Éléments finis** La méthode des éléments finis se base sur la forme faible de l'équation d'onde, c'est-à-dire la forme intégrale utilisant une fonction d'essai. Selon celle-ci, la solution intégrale est approximée en divisant le domaine spatial en éléments sur lesquels des fonctions de base sont définies. La forme et l'extension des éléments ainsi que le choix de la base différencient les types de méthodes par éléments finis. Notons particulièrement la méthode aux éléments finis continus (Marfurt, 1984; Koketsu *et al.*, 2004), la méthode de Galerkin discontinue (Käser & Dumbser, 2006; de la Puente *et al.*, 2007), et la méthode des éléments spectraux (Faccioli *et al.*, 1997; Komatitsch & Tromp, 1999; Komatitsch *et al.*, 2000). Ces méthodes ont l'avantage d'utiliser une grille qui peut s'adapter aux discontinuités de l'espace à discrétiser. En particulier, la surface libre est automatiquement implémentée très précisément par ces méthodes. Cependant, l'utilisation de grilles irrégulières requiert la création d'un maillage de qualité, ce qui peut parfois être difficile, particulièrement pour la méthode des éléments spectraux qui se base sur des éléments parallélépipèdes.

- **Différences finies en fréquence** La méthode des différences finies en fréquence utilise l'équation d'onde en fréquence (équation (2.26)). Le domaine spatial est discrétisé selon une grille régulière, permettant d'approximer les dérivées spatiales par la différence entre les éléments adjacents de la grille. La résolution de la fonction d'onde s'obtient alors par l'inversion d'une large matrice \mathbf{F} , ce qui peut se faire par des méthodes directes (Marfurt, 1984; Stekl & Pratt, 1998; Operto *et al.*, 2007) ou itératives (Riyanti *et al.*, 2006; Plessix, 2009). Une fois que l'inverse de cette matrice est obtenu, la solution pour un grand nombre de sources peut être calculée par un simple produit matriciel, ce qui est d'une très grande efficacité. De plus, comme l'inversion de formes d'ondes complètes peut être effectuée en utilisant qu'un nombre restreint de fréquences (Bunks *et al.*, 1995; Sirgue & Pratt, 2004), il est possible de n'inverser cette matrice que pour les fréquences sélectionnées. À noter que pour les problèmes en 3D, la taille de la matrice \mathbf{F} explose et l'approche en fréquence devient prohibitive.

- **Méthodes pseudo-spectrales** La méthode pseudo-spectrale (Fornberg, 1987, 1988; Fomel *et al.*, 2013), à l'instar des méthodes en fréquence, utilise la transformée de Fourier pour transformer les dérivées en produits dans le domaine de Fourier, cette fois-ci pour les dérivées spatiales. Les dérivées temporelles sont au contraire approximées par des différences finies, généralement d'ordre deux, permettant une relation de récurrence explicite entre les pas de temps. Étant donné la précision de la base de Fourier pour les signaux oscillatoires, un très grand pas spatial peut être choisi sans causer de dispersion. Cette limite peut approcher 2, soit la limite du théorème d'échantillonnage. Comme le schéma temporel est explicite, une transformée directe et une transformée inverse par pas de temps doivent être effectuées, ce qui peut être coûteux, surtout en 3D. Notons aussi que l'utilisation du domaine de Fourier en temps et en fréquence n'est pas possible aujourd'hui en raison de la grande largeur de bande de la matrice à inverser dans ce cas.

- **Différences finies en temps** La méthode des différences finies dans le domaine temporel approxime tant les dérivées spatiales que temporelles par la différence d'éléments adjacents sur une grille régulière (Alterman & Karal, 1968; Levander, 1988; Virieux, 1986). En utilisant une différence finie d'ordre deux dans le temps, il est possible d'obtenir une relation de récurrence n'impliquant que le pas de temps précédents. Ainsi, contrairement aux différences finies en fréquence, l'inversion de la matrice \mathbf{F} n'est pas requise dans le temps. De plus, la

solution pour toutes les fréquences est obtenue simultanément. Cependant, comme toutes les méthodes se basant sur des grilles cartésiennes régulières, les différences finies dans le temps ont le désavantage de difficilement reproduire une surface libre de forme arbitraire.

Dans ce travail, la méthode des différences finies dans le temps sera étudiée plus en détail et une implémentation parallèle efficace est développée au Chapitre 3. La méthode des différences finies dans le temps a été choisie pour sa grande simplicité d'implémentation, sa capacité d'être parallélisée qui est supérieure aux autres méthodes, et la possibilité d'effectuer des simulations en 3D avec les moyens numériques actuels. Le reste de cette section porte sur quelques notions élémentaires de cette méthode.

2.2.2 Fondements théoriques

À la base, il est possible de justifier l'approximation des différences finies par un développement en série de Taylor. Par exemple, utilisant un pas Δx , il est possible d'écrire

$$v(x + \Delta x) = v(x) + \Delta x \partial_x v(x) + \frac{\Delta x^2}{2} \partial_{xx} v(x) + \dots \quad (2.35)$$

et donc

$$\frac{v(x + \Delta x) - v(x)}{\Delta x} = \partial_x v(x) + \mathcal{O}(\Delta x), \quad (2.36)$$

ce qui correspond aux différences avant d'ordre 1. Les différences arrière sont obtenues de la même façon, avec une perturbation Δx négative, et la moyenne de ces deux opérateurs donne les différences centrées. En généralisant cette démarche, il est possible d'obtenir un opérateur centré en différences finies d'ordre $2N_{DF}$ donné par (Fornberg, 1987)

$$\partial_x v(x) \approx Dv(x) = \sum_{n=-N_{DF}}^{N_{DF}} g_{n,N_{DF}} v(x - n\Delta x), \quad (2.37a)$$

$$g_{n,N_{DF}} = \begin{cases} \frac{(N_{DF}!)^2 (-1)^{n+1}}{n(n + N_{DF})!(N_{DF} - n)!\Delta x} & \text{si } n \neq 0, \\ 0 & \text{si } n = 0. \end{cases} \quad (2.37b)$$

Plus l'ordre $2N$ est élevé, plus l'approximation de la dérivée est précise, mais plus le nombre de points utilisés dans le calcul est grand. Un ordre plus élevé permet en général d'augmenter le pas de discrétisation, mais peut entraîner des erreurs de modélisation lorsque les paramètres élastiques varient rapidement. Habituellement, un ordre $O(4)$ est utilisé en modélisation sismique. Le cas limite, lorsque N_{DF} tend vers l'infini, produit la méthode pseudo-spectrale (Fornberg, 1987). Notons aussi que les coefficients $g_{n,N}$ peuvent être obtenus selon d'autres approches que la série de Taylor. Notamment, ces coefficients peuvent être choisis pour réduire la dispersion selon la méthode de Holberg (1987), ce qui permet d'augmenter le pas de discrétisation.

2.2.3 Grilles décalées

Les premières implémentations des différences finies de l'équation d'onde mécanique étaient basées sur des grilles régulières où toutes les propriétés physiques et les champs d'ondes étaient définis aux mêmes nœuds (Alterman & Karal, 1968; Alford *et al.*, 1974; Kelly *et al.*, 1976). L'introduction de grilles décalées (Virieux, 1984, 1986; Levander, 1988) réduit l'espacement effectif des nœuds de la grille, ce qui améliore grandement la précision de l'approximation de la dérivée. Pour comprendre l'avantage des grilles décalées, nous suivons le développement présenté par Fichtner (2011) et Igel *et al.* (1995). Ce développement se base l'approximation à bande limitée de l'opérateur différentiel. En effet, la différentiation dans le domaine spatial ∂_x représente une multiplication par ik_x dans le domaine de Fourier. Lorsque l'espace est discrétisé, cet opérateur devient à bande limitée avec un nombre d'onde maximal égal au nombre d'onde de Nyquist $k_{max} = \pi/\Delta x$. Plus précisément,

$$\begin{aligned}\partial_x v(x) &= F^{-1}(ik\hat{v}(k)) \\ &\approx F^{-1}(ik [H(k + k_{max}) + H(k - k_{max})] \hat{v}),\end{aligned}\tag{2.38}$$

ce qui représente une convolution dans le temps, c'est-à-dire

$$\begin{aligned}\partial_x v(x) &\approx \frac{k_{max}}{\pi} \int_{-\infty}^{\infty} \frac{d}{dx'} \text{sinc}(k_{max}x') v(x - x') dx' \\ &\approx \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{1}{x'^2} [k_{max}x' \cos(k_{max}x') - \sin(k_{max}x')] v(x - x') dx' .\end{aligned}\tag{2.39}$$

Pour une discrétisation de pas Δx , $x' = n\Delta x$ et la convolution devient une somme:

$$\partial_x v(x) \approx \sum_{n=-N_{DF}}^{N_{DF}} g_{n,N_{DF}} v(x - n\Delta x), \quad (2.40a)$$

$$g_{n,N_{DF}} = \begin{cases} \frac{(-1)^n}{n\Delta x} & \text{si } n \neq 0, \\ 0 & \text{si } n = 0. \end{cases} \quad (2.40b)$$

À noter que ces coefficients sont différents des coefficients de Taylor de la section précédente en raison de la méthode de dérivation différente. Dans le cas d'une grille décalée, la position est plutôt donnée par $x' = (n + 1/2)\Delta x$, ce qui donne les formules suivantes:

$$\partial_x v(x) \approx \sum_{n=-N_{DF}}^{N_{DF}-1} g_{n,N_{DF}} v(x - (n + 1/2)\Delta x), \quad (2.41a)$$

$$g_{n,N_{DF}} = \frac{(-1)^{n+1}}{\pi(n + 1/2)^2 \Delta x}. \quad (2.41b)$$

Il est possible d'évaluer la performance des ces approximations en comparant leur réponse en fréquence à la réponse analytique, c'est-à-dire $F[\partial_x] = ik_x$. Ainsi, la réponse en fréquence des convolutions données aux relations (2.37), (2.40) et (2.41) doit être imaginaire et proportionnelle au nombre d'onde. Ceci est testé à la Figure 2.6. Pour des opérateurs à 4, 8 et 16 points, la convolution basée sur les grilles décalées est beaucoup plus performante. La grille standard avec les coefficients de l'équation (2.40) a une réponse qui oscille autour de la solution tandis que les coefficients de Taylor ne sont valides que pour des nombres d'onde inférieurs à 50 % du nombre d'onde de Nyquist. La meilleure performance de la grille décalée peut s'expliquer par l'échantillonnage du noyau de convolution exact de l'équation (2.38), soit $\frac{d}{dx} \text{sinc}(k_{max}x)$. Ainsi, comme le montre la Figure 2.6d, la valeur des points échantillonnés diminue beaucoup plus rapidement pour la grille décalée. Ainsi, la convolution continue peut être bien approximée par une somme discrète n'incluant que les premiers termes. Comme cette somme diminue plus lentement pour la grille conventionnelle, un plus grand nombre de termes est requis pour obtenir la même précision.

2.2.4 Dispersion et stabilité

Comme le montre la section précédente, la discrétisation des dérivées spatiales et temporelles entraîne des erreurs numériques. Ces erreurs se manifestent par une dispersion numérique et/ou

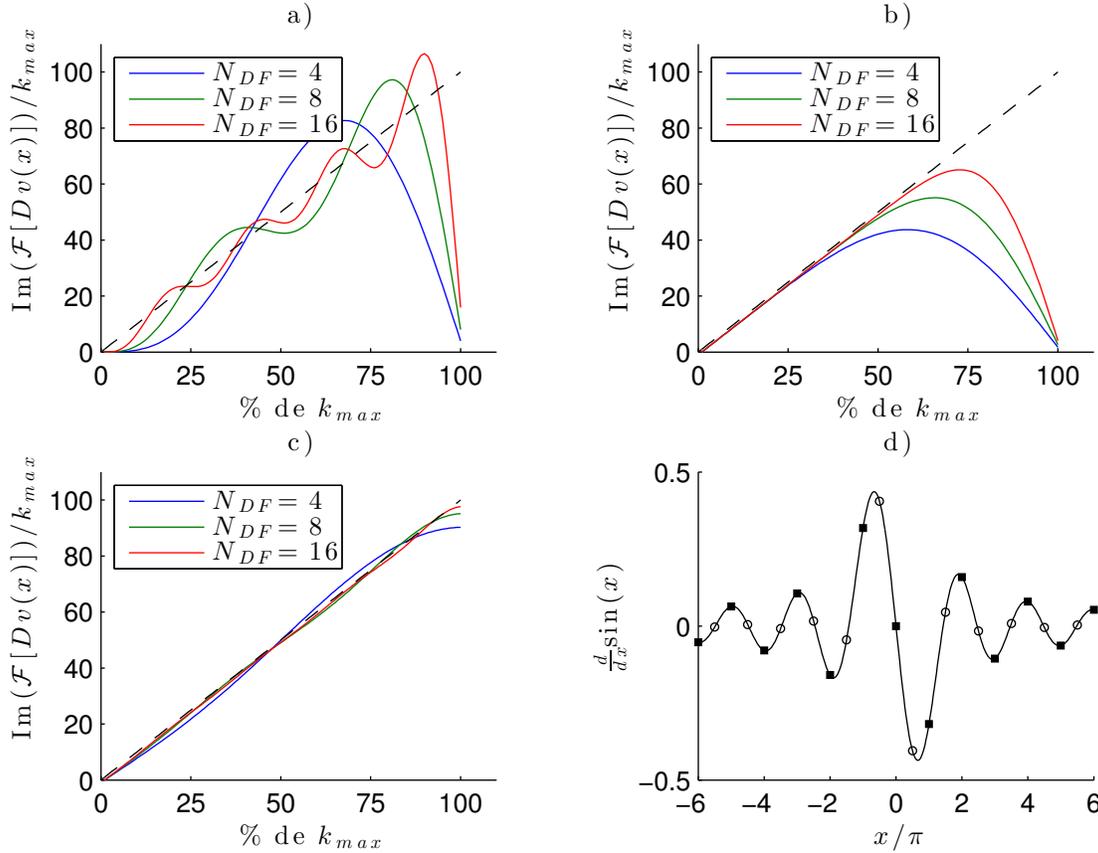


Figure 2.6 – Réponse en fréquence des opérateurs en différences finies: a) grille standard (coefficients éq. (2.40)), b) coefficients de Taylor en grille standard, c) grille décalée (coefficients éq. (2.41)). En d), le noyau exact de convolution est montré avec les points d'échantillonnage de la grille standard (cercles ouverts) et de la grille décalée (carrés pleins).

l'instabilité de la solution. Une analyse standard pour développer les critères de stabilité est l'analyse de von Neumann (Crank & Nicolson, 1947; Charney *et al.*, 1950). Mentionnons l'article de Marfurt (1984), qui fait une bonne révision de la précision des différentes méthodes de résolution numérique de l'équation d'onde, et Moczo (2000), qui procède à l'analyse de von Neumann pour la formulation à grilles décalées de l'équation d'onde élastique 3D.

Un milieu dispersif est un milieu pour lequel la vitesse de phase dépend de la fréquence. Par conséquent, l'onde se déforme lorsqu'elle se propage. Pour un milieu homogène élastique, les ondes P et S ne sont pas dispersives en théorie. Cependant, la discrétisation des dérivés partielles entraîne une dispersion non physique, devant être minimisée. L'analyse de von Neumann fournit un critère précis afin de minimiser la dispersion numérique, soit

$$\Delta x \leq K_d \frac{V_{min}}{f_{max}}, \quad (2.42)$$

où la constante K_d dépend de la dimension de l'équation d'onde, de l'opérateur des différences finies et d'une tolérance à l'erreur. Un pas de temps spatial maximal peut donc être établi à l'aide de ce critère.

Il est aussi possible d'établir un critère de stabilité de la solution en différences finies, qui est conditionnellement stable. Une solution est stable lorsqu'une erreur numérique ne grandit pas au cours des itérations. La condition de stabilité, ou condition CFL nommée d'après R. Courant, K. Friedrichs et H. Lewy (Courant *et al.*, 1928), s'exprime comme

$$\Delta t \leq K_s \frac{\Delta x}{V_{max}}. \quad (2.43)$$

La constante K_s est propre à la dimension du problème et au type de grille utilisée. Les critères de dispersion et de CFL permettent de dimensionner la grille ainsi que le pas de temps d'une simulation aux différences finies. À noter que plus le pas spatial est petit, plus le pas temporel doit aussi être réduit. Pour de grands contrastes de vitesse, par exemple pour de faibles vitesses de cisaillement et des vitesses des ondes P élevées, la grandeur du pas temporel peut devenir très faible, menant à des coûts de calcul élevés.

2.2.5 Équation d'onde viscoélastique isotrope en 3D avec grilles décalées

La formulation en différences finies de l'équation en trois dimensions viscoélastique isotrope (équation (2.29)) est proposée par Bohlen (2002). Les principaux résultats sont ici résumés, les algorithmes des chapitres suivants étant basés sur cette solution.

Cohérence des grilles décalées

Nous utilisons des grilles décalées dans le temps et dans l'espace. Spatialement, certaines variables sont ainsi décalées d'un demi pas spatial et d'autres non. Deux opérateurs en différences finies sont ainsi nécessaires, les opérateurs avant et arrière, respectivement données par

$$D^+ v^p \approx \sum_{n=0}^{N_{DF}-1} g_{n,N_{DF}} (v^{p+n+1} - v^{p-n}), \quad (2.44a)$$

$$D^- v^p \approx \sum_{n=0}^{N_{DF}-1} g_{n,N_{DF}} (v^{p+n} - v^{p-n-1}). \quad (2.44b)$$

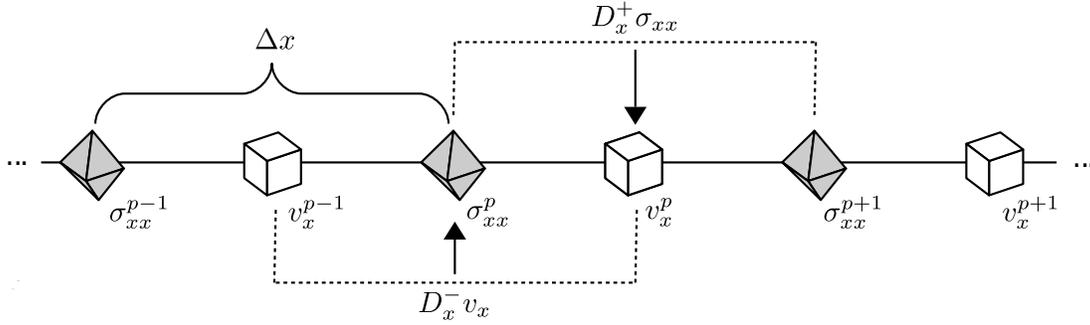


Figure 2.7 – Position des dérivées spatiales avant et arrière pour un schéma en différences finies d'ordre 2.

La différence arrière est équivalente à l'expression (2.41) de la dérivée sur grille décalée en 1D, c'est-à-dire décalée de $\Delta x/2$. La différence avant cependant correspond à une différence décalée de $-\Delta x/2$. Le choix de l'opérateur avant ou arrière doit être cohérent avec la position des noeuds des variables. En effet, comme le montre la Figure 2.7, la position de la dérivée se situe au centre du patron de l'opérateur en différences finies. Par exemple, la dérivée spatiale de v_x sur ce schéma est disponible au noeud de σ_{xx} avec l'opérateur avant. Il est donc possible d'utiliser cette dérivée lors de la mise à jour de σ_{xx} . De même, la dérivée spatiale de σ_{xx} est disponible au noeud décalé de v_x avec l'opérateur arrière, ce qui permet de mettre à jour v_x . De façon générale, la grille décalée doit être cohérente pour toutes les variables, c'est-à-dire qu'une variable et les dérivées spatiales impliquées dans sa mise à jour doivent être définies à la même position.

Les dérivées temporelles sont aussi décalées dans le temps: les vitesses et les variables de mémoire sont définies aux multiples de Δt tandis que les contraintes sont décalées de $\Delta t/2$. Plus précisément, les dérivées dans le temps des contraintes et des vitesses sont discrétisées par les différences finies d'ordre 2 suivantes

$$\partial_t v((t + 1/2)\Delta t) \approx \frac{v^{t+1} - v^t}{\Delta t}, \quad (2.45)$$

$$\partial_t \sigma(t\Delta t) \approx \frac{\sigma^{t+1/2} - \sigma^{t-1/2}}{\Delta t}. \quad (2.46)$$

Les variables de mémoires sont quant à elles mises à jour par la méthode de Crank–Nicolson, exprimée comme

$$\frac{r^{t+1} - r^t}{\Delta t} = \frac{1}{2} \left(F(r)^{t+1} + F(r)^t \right), \quad (2.47)$$

où $F(r)^{t+1}$ est l'équation différentielle donnant la variation dans le temps de la variable de mémoire évaluée au temps $(t+1)\Delta t$. Ainsi, les dérivées temporelles des vitesses sont définies pour un temps décalé de $\Delta t/2$, c'est-à-dire aux mêmes instants que les variables des contraintes. Pareillement, les dérivées temporelles des contraintes sont définies aux mêmes instants que les vitesses. Bref, à l'instar des dérivées spatiales, les dérivées temporelles de la grille décalée doivent être utilisées de façon cohérente pour discrétiser les équations partielles différentielles de l'équation (2.29).

Cellule élémentaire

Les positions relatives de la grille de chacune des variables doivent être spécifiées pour résoudre le système d'équations différentielles partielles par différences finies. Ces positions sont totalement spécifiées par la cellule élémentaire, présentée à la Figure 2.8. Celle-ci est construite en définissant trois noeuds décalés pour les vitesses

$$v_x^{p,q,r,t} = v_x((p+1/2)\Delta x, q\Delta y, r\Delta z, t\Delta t), \quad (2.48a)$$

$$v_y^{p,q,r,t} = v_y(p\Delta x, (q+1/2)\Delta y, r\Delta z, t\Delta t), \quad (2.48b)$$

$$v_z^{p,q,r,t} = v_z(p\Delta x, q\Delta y, (r+1/2)\Delta z, t\Delta t). \quad (2.48c)$$

En d'autres mots, chaque composante des vitesses est décalée de $\Delta x/2$ dans la direction respective de sa composante (v_x en x , etc), pour un temps non décalé ($t\Delta t$). Les contraintes et les variables de mémoire sont, quant à elles, discrétisées selon les positions suivantes:

$$\sigma_{xx}^{p,q,r,t+1/2} = \sigma_{xx}(p\Delta x, q\Delta y, r\Delta z, (t+1/2)\Delta t), \quad (2.49a)$$

$$\sigma_{yy}^{p,q,r,t+1/2} = \sigma_{yy}(p\Delta x, q\Delta y, r\Delta z, (t+1/2)\Delta t), \quad (2.49b)$$

$$\sigma_{zz}^{p,q,r,t+1/2} = \sigma_{zz}(p\Delta x, q\Delta y, r\Delta z, (t+1/2)\Delta t), \quad (2.49c)$$

$$\sigma_{yz}^{p,q,r,t+1/2} = \sigma_{yz}(p\Delta x, (q+1/2)\Delta y, (r+1/2)\Delta z, (t+1/2)\Delta t), \quad (2.49d)$$

$$\sigma_{xz}^{p,q,r,t+1/2} = \sigma_{xz}((p+1/2)\Delta x, q\Delta y, (r+1/2)\Delta z, (t+1/2)\Delta t), \quad (2.49e)$$

$$\sigma_{xy}^{p,q,r,t+1/2} = \sigma_{xy}((p+1/2)\Delta x, (q+1/2)\Delta y, r\Delta z, (t+1/2)\Delta t), \quad (2.49f)$$

$$r_{xx}^{p,q,r,t} = r_{xx}(p\Delta x, q\Delta y, r\Delta z, t\Delta t), \quad (2.49g)$$

$$r_{yy}^{p,q,r,t} = r_{yy}(p\Delta x, q\Delta y, r\Delta z, t\Delta t), \quad (2.49h)$$

$$r_{zz}^{p,q,r,t} = r_{zz}(p\Delta x, q\Delta y, r\Delta z, t\Delta t), \quad (2.49i)$$

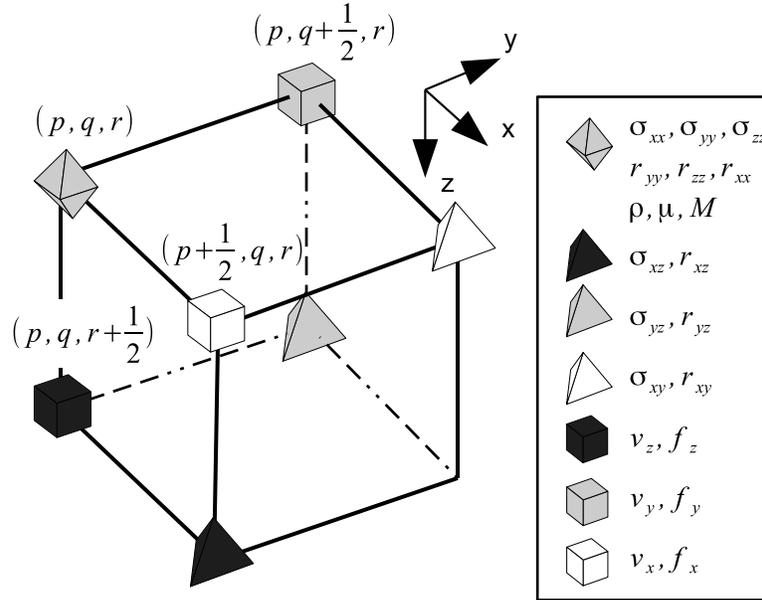


Figure 2.8 – Cellule élémentaire de la grille standard décalée utilisée pour discrétiser l'équation 3D viscoélastique isotrope.

$$r_{yz}^{p,q,r,t} = r_{yz}(p\Delta x, (q + 1/2)\Delta y, (r + 1/2)\Delta z, t\Delta t), \quad (2.49j)$$

$$r_{xz}^{p,q,r,t} = r_{xz}((p + 1/2)\Delta x, q\Delta y, (r + 1/2)\Delta z, t\Delta t), \quad (2.49k)$$

$$r_{xy}^{p,q,r,t} = r_{xy}((p + 1/2)\Delta x, (q + 1/2)\Delta y, r\Delta z, t\Delta t). \quad (2.49l)$$

À remarquer que chaque variable de mémoire est définie à la même position spatiale que la contrainte correspondante, mais est décalée dans le temps. Les contraintes normales sont définies au noeud d'origine de la cellule élémentaire de la grille élémentaire. En bref, la cellule élémentaire résume le décalage de la grille de chacune des variables (Figure 2.8).

Discrétisation des équations différentielles partielles

Avec ce patron de grilles décalées en temps et dans l'espace, et remplaçant les dérivées par les opérateurs avant et arrière appropriés dans les équations (2.29), il est possible d'obtenir des

équations de récurrence explicites pour chacune des variables. Pour les contraintes, nous obtenons

$$\begin{aligned} \sigma_{xx}^{p,q,r,t+1/2} &= \sigma_{xx}^{p,q,r,t-1/2} + \Delta t M^0{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) \\ &\quad - \Delta t 2\mu^0{}^{p,q,r} (D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{xx}^l{}^{p,q,r,t} + r_{xx}^l{}^{p,q,r,t+1} \right) \\ &\quad + s_{\sigma_{xx}}^{p,q,r,t+1/2}, \end{aligned} \quad (2.50a)$$

$$\begin{aligned} \sigma_{yy}^{p,q,r,t+1/2} &= \sigma_{yy}^{p,q,r,t-1/2} + \Delta t M^0{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) \\ &\quad - \Delta t 2\mu^0{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{yy}^l{}^{p,q,r,t} + r_{yy}^l{}^{p,q,r,t+1} \right) \\ &\quad + s_{\sigma_{yy}}^{p,q,r,t+1/2}, \end{aligned} \quad (2.50b)$$

$$\begin{aligned} \sigma_{zz}^{p,q,r,t+1/2} &= \sigma_{zz}^{p,q,r,t-1/2} + \Delta t M^0{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) \\ &\quad - \Delta t 2\mu^0{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t}) + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{zz}^l{}^{p,q,r,t} + r_{zz}^l{}^{p,q,r,t+1} \right) \\ &\quad + s_{\sigma_{zz}}^{p,q,r,t+1/2}, \end{aligned} \quad (2.50c)$$

$$\begin{aligned} \sigma_{yz}^{p,q,r,t+1/2} &= \sigma_{yz}^{p,q,r,t-1/2} + \Delta t \mu_{yz}^0{}^{p,q,r} (D_y^+ v_z^{p,q,r,t} + D_z^+ v_y^{p,q,r,t}) \\ &\quad + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{yz}^l{}^{p,q,r,t} + r_{yz}^l{}^{p,q,r,t+1} \right) + s_{\sigma_{yz}}^{p,q,r,t+1/2}, \end{aligned} \quad (2.50d)$$

$$\begin{aligned} \sigma_{xz}^{p,q,r,t+1/2} &= \sigma_{xz}^{p,q,r,t-1/2} + \Delta t \mu_{xz}^0{}^{p,q,r} (D_x^+ v_z^{p,q,r,t} + D_z^+ v_x^{p,q,r,t}) \\ &\quad + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{xz}^l{}^{p,q,r,t} + r_{xz}^l{}^{p,q,r,t+1} \right) + s_{\sigma_{xz}}^{p,q,r,t+1/2}, \end{aligned} \quad (2.50e)$$

$$\begin{aligned} \sigma_{xy}^{p,q,r,t+1/2} &= \sigma_{xy}^{p,q,r,t-1/2} + \Delta t \mu_{xy}^0{}^{p,q,r} (D_x^+ v_y^{p,q,r,t} + D_y^+ v_x^{p,q,r,t}) \\ &\quad + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{xy}^l{}^{p,q,r,t} + r_{xy}^l{}^{p,q,r,t+1} \right) + s_{\sigma_{xy}}^{p,q,r,t+1/2}, \end{aligned} \quad (2.50f)$$

où nous avons utilisé le module des ondes P : $M = \lambda + 2\mu$. On peut faire de même pour les équations des variables de mémoire des équations (2.24), ce qui donne

$$\begin{aligned} r_{xx}^l{}^{p,q,r,t+1} &= \left(1 - \frac{1}{2\tau_{\sigma l}} \right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} M^l{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) \right. \\ &\quad \left. - \frac{\Delta t}{\tau_{\sigma l}} 2\mu^l{}^{p,q,r} (D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) + \left(1 + \frac{1}{2\tau_{\sigma l}} \right) r_{xx}^l{}^{p,q,r,t} \right], \end{aligned} \quad (2.51a)$$

$$\begin{aligned} r_{yy}^l{}^{p,q,r,t+1} &= \left(1 - \frac{1}{2\tau_{\sigma l}} \right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} M^l{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) \right. \\ &\quad \left. - \frac{\Delta t}{\tau_{\sigma l}} 2\mu^l{}^{p,q,r} (D_x^- v_x^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) + \left(1 + \frac{1}{2\tau_{\sigma l}} \right) r_{yy}^l{}^{p,q,r,t} \right], \end{aligned} \quad (2.51b)$$

$$r_{zz}^{l\ p,q,r,t+1} = \left(1 - \frac{1}{2\tau_{\sigma l}}\right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} M^{l\ p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} + D_z^- v_z^{p,q,r,t}) - \frac{\Delta t}{\tau_{\sigma l}} 2\mu^{l\ p,q,r} (D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t}) + \left(1 + \frac{1}{2\tau_{\sigma l}}\right) r_{zz}^{l\ p,q,r,t} \right], \quad (2.51c)$$

$$r_{yz}^{l\ p,q,r,t+1} = \left(1 - \frac{1}{2\tau_{\sigma l}}\right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} \mu_{yz}^{l\ p,q,r} (D_y^+ v_z^{p,q,r,t} + D_z^+ v_y^{p,q,r,t}) + \left(1 + \frac{1}{2\tau_{\sigma l}}\right) r_{yz}^{l\ p,q,r,t} \right], \quad (2.51d)$$

$$r_{xz}^{l\ p,q,r,t+1} = \left(1 - \frac{1}{2\tau_{\sigma l}}\right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} \mu_{xz}^{l\ p,q,r} (D_x^+ v_z^{p,q,r,t} + D_z^+ v_x^{p,q,r,t}) + \left(1 + \frac{1}{2\tau_{\sigma l}}\right) r_{xz}^{l\ p,q,r,t} \right], \quad (2.51e)$$

$$r_{xy}^{l\ p,q,r,t+1} = \left(1 - \frac{1}{2\tau_{\sigma l}}\right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} \mu_{xy}^{l\ p,q,r} (D_x^+ v_y^{p,q,r,t} + D_y^+ v_x^{p,q,r,t}) + \left(1 + \frac{1}{2\tau_{\sigma l}}\right) r_{xy}^{l\ p,q,r,t} \right]. \quad (2.51f)$$

Enfin, les équations pour la mise à jour des vitesses sont données par:

$$v_x^{p,q,r,t+1} = v_x^{p,q,r,t} + \frac{\Delta t}{\rho_x^{p,q,r}} \left[D_x^+ \sigma_{xx}^{p,q,r,t-1/2} + D_y^- \sigma_{xy}^{p,q,r,t-1/2} + D_z^- \sigma_{xz}^{p,q,r,t-1/2} \right] + s_{v_x}^{p,q,r,t}, \quad (2.52a)$$

$$v_y^{p,q,r,t+1} = v_y^{p,q,r,t} + \frac{\Delta t}{\rho_y^{p,q,r}} \left[D_y^+ \sigma_{yy}^{p,q,r,t-1/2} + D_x^- \sigma_{xy}^{p,q,r,t-1/2} + D_z^- \sigma_{yz}^{p,q,r,t-1/2} \right] + s_{v_y}^{p,q,r,t}, \quad (2.52b)$$

$$v_z^{p,q,r,t+1} = v_z^{p,q,r,t} + \frac{\Delta t}{\rho_z^{p,q,r}} \left[D_z^+ \sigma_{zz}^{p,q,r,t-1/2} + D_y^- \sigma_{yz}^{p,q,r,t-1/2} + D_x^- \sigma_{xz}^{p,q,r,t-1/2} \right] + s_{v_z}^{p,q,r,t}. \quad (2.52c)$$

Bref, le choix du positionnement des noeuds de chaque variable dans la cellule élémentaire de la Figure 2.8 mène à des équations de mises à jour parfaitement cohérentes. Le lecteur peut facilement vérifier que toutes les dérivées spatiales sont définies au même noeud que la variable mise à jour. À noter que pour un milieu anisotrope, ce n'est plus le cas et les dérivées doivent être interpolées (Igel *et al.*, 1995), ou une grille alternative est requise (Tam & Webb, 1993; Saenger & Bohlen, 2004).

Estimation des paramètres viscoélastiques aux noeuds décalés

Étant donné que les variables sismiques sont définies à différentes positions de la cellule élémentaire, les paramètres élastiques doivent être interpolés aux positions des différentes variables. Moczo (2002) montrent que la densité et les niveaux d'atténuation des ondes S doivent être moyennés arithmétiquement, tandis le module de cisaillement doit être moyenné de façon harmonique. Par exemple, la densité moyennée en x est

$$\rho_x^{p,q,r} = \frac{\rho^{p,q,r} + \rho^{p+1,q,r}}{2}, \quad (2.53)$$

et le module de cisaillement moyenné en xz vaut

$$\mu_{xz}^{p,q,r} = \frac{4}{\mu^{p,q,r} + \mu^{p+1,q,r} + \mu^{p,q,r+1} + \mu^{p+1,q,r+1}}. \quad (2.54)$$

Termes sources

La résolution par différences finies permet de facilement implanter un terme source ponctuel, pour différentes formes de sources. Des forces directionnelles en x , y ou z sont simulées en additionnant au noeud correspondant à la position de la source les termes $s_{v_x}^{p,q,r,t}$, $s_{v_y}^{p,q,r,t}$ et $s_{v_z}^{p,q,r,t}$, respectivement. Par exemple, une source à impact est bien modélisée par une force directionnelle. De même, une source explosive est facilement implantée en additionnant au noeud correspondant à la position de l'explosion les sources des contraintes normales $s_{\sigma_{xx}}^{p,q,r,t+1/2}$, $s_{\sigma_{yy}}^{p,q,r,t+1/2}$, $s_{\sigma_{zz}}^{p,q,r,t+1/2}$, qui doivent être égales dans ce cas. Plusieurs formes de signal peuvent être choisies pour ces termes sources, par exemple l'onde de Ricker:

$$s_{ricker} = (1 - 2\pi^2 f^2 t^2) e^{-\pi^2 f^2 t^2}. \quad (2.55)$$

À noter que la forme de ce terme source devient un paramètre inconnu lors de l'inversion de formes d'ondes, et doit être estimée lors de l'inversion (voir Pratt (1999) sur ce sujet). De plus, lorsque la source n'est pas positionnée exactement à un noeud de la grille, le terme source doit être interpolé aux noeuds adjacents en utilisant une approximation à bande limitée de la fonction delta (Hicks, 2002).

Démarche de mise à jour

En résumé, l'équation d'onde peut être résolue avec les équations (2.50), (2.51) et (2.52) en suivant la procédure suivante:

1. **Mise à jour des vitesses** Pour chaque noeud de la grille, les dérivées spatiales des contraintes $D_i^\pm \sigma_{jk}^{p,q,r,t-1/2}$ sont calculées et utilisées pour passer de v_i^t à v_i^{t+1} selon les équations (2.52). Chaque noeud correspondant à une position de source directionnelle est additionné du terme source au temps t , $s_{v_i}^{p,q,r,t}$.

2. **Mise à jour partielle des contraintes** Pour chaque noeud de la grille, les dérivées spatiales des vitesses $D_i^+ v_j^{p,q,r,t}$ sont calculées. Les contraintes $\sigma_{ij}^{t+1/2}$ sont ensuite partiellement mises à jour selon les équations (2.50), en omettant pour le moment les termes $r_{ij}^{l,p,q,r,t+1}$, car encore inaccessibles. Chaque noeud correspondant à une position de source explosive est additionné du terme source au temps $t + 1/2$, $s_{\sigma_{ii}}^{p,q,r,t}$.
3. **Mise à jour des variables de mémoire** À partir des dérivées spatiales $D_m^+ v_i^{p,q,r,t}$ calculées à l'étape précédente, les variables de mémoire sont mises à jour, passant de $r_{ij}^{l,p,q,r,t}$ à $r_{ij}^{l,p,q,r,t+1}$ selon l'équation (2.51).
4. **Mise à jour complète des contraintes** Les contraintes sont finalement totalement mises à jour en ajoutant le dernier terme de l'équation (2.50), $r_{ij}^{l,p,q,r,t+1}$, obtenu à l'étape précédente.

Cette procédure est itérée à partir du temps 0 pour un nombre de pas de temps N_t . Comme les variables ne sont requises que pour un seul pas de temps à la fois, la mémoire allouée pour les $9 + 6L$ variables est réutilisée pour chaque pas de temps. En d'autres mots, la mise à jour d'une variable écrase la variable au pas de temps précédent.

2.2.6 Conditions aux limites

Condition initiale

La résolution par différences finies de l'équation d'onde requiert des conditions limites. En particulier, l'itération débute au temps 0, et nous supposons le milieu initialement au repos, c'est-à-dire

$$v_i^{p,q,r,t=0} = \sigma_{ij}^{p,q,r,t=0} = r_{ij}^{l,p,q,r,t=0} = 0. \quad (2.56)$$

Conditions aux frontières

De plus, comme les simulations numériques occupent un espace mémoire limité, un domaine spatial de dimensions finies doit être défini. Le plus simple est d'imposer des contraintes et des vitesses nulles aux frontières du domaine, c'est-à-dire

$$v_i^{\Gamma,t} = \sigma_{ij}^{\Gamma,t} = r_{ij}^{l,\Gamma,t} = 0, \quad (2.57)$$

où Γ correspond à la frontière spatiale du domaine de simulation. Cependant, cette condition frontière est peu réaliste pour des levés sismiques de faibles dimensions pour lesquels le sous-sol est mieux représenté par un milieu semi-infini. En effet, ces conditions rigides causeront des réflexions aux bordures du modèle qui ne seront pas présentes dans les enregistrements. Pour simuler un milieu semi-infini, une des méthodes les plus efficaces est celle des bordures à convolution parfaitement adaptée (CPML) proposée par (Roden & Gedney, 2000) et adaptée pour l'équation d'onde élastique par (Komatitsch & Martin, 2007). Selon cette méthode, les dérivées spatiales de chaque variable sont corrigées dans la zone atténuante selon:

$$\tilde{D}_i v_j^t = \frac{1}{k_i} D_i v_j^t + \psi_{\partial_i v_j}^t, \quad (2.58)$$

avec

$$\psi_{\partial_i v_j}^t = b_i \psi_{\partial_i v_j}^{t-1} + a_i \frac{1}{k_i} D_i v_j^t, \quad (2.59)$$

et

$$a_i = \frac{d_i}{\kappa_i(d_i + \kappa_i \alpha_i)} (b_i - 1), \quad (2.60)$$

$$b_i = e^{-(d_i/\kappa_i + \alpha_i)\Delta t}. \quad (2.61)$$

Les valeurs des coefficients α_i , κ_i et d_i peuvent varier d'un auteur à l'autre. Nous suivons (Komatitsch & Martin, 2007) et adoptons

$$d_i = d_0 \left(\frac{y}{L}\right)^2, \quad (2.62)$$

$$\alpha_i = \pi f_0 \left(1 - \frac{x}{L}\right), \quad (2.63)$$

$$\kappa_i = 1, \quad (2.64)$$

où L est ici la largeur de la bande absorbante.

Surface libre

Enfin, pour modéliser la surface du sol, les conditions de surface libre doivent être implantées. Une surface libre répond aux conditions suivantes:

$$n_j \sigma_{ij} = 0, \quad (2.65a)$$

$$n_j r_{ij}^l = 0, \quad (2.65b)$$

où n_i est le vecteur unitaire normal à la surface libre. Le principal problème pour implanter cette condition dans les algorithmes de différences finies est le calcul de la dérivée qui nécessite plusieurs points adjacents. Or, ceci signifie que pour les cellules à proximité de la surface libre, des points au-dessus de celle-ci sont requis. Plusieurs méthodes ont été proposées pour résoudre ce problème. Notons la méthode du vacuum (Zahradnik & Urban, 1984; Ohminato & Chouet, 1997; Bohlen & Saenger, 2006) pour laquelle les coefficients du tenseur de rigidité sont fixés à 0 au-dessus de la surface libre. Cette méthode a le mérite d'être simple d'implémentation et de permettre de modéliser la topographie. Cependant, sa précision est limitée et une grille excessivement fine, d'au moins 60 points par longueur d'onde pour une surface libre non-horizontale (Bohlen & Saenger, 2006). Les méthodes internes modifient le schéma de différences finies autour de la surface libre pour éviter d'avoir à utiliser des points au-dessus de la surface (Kristek *et al.*, 2002; Moczo *et al.*, 2004). Notons aussi l'approche du raffinement de la grille près de la surface libre (Kristek *et al.*, 2010), l'utilisation d'une grille déformée pour accommoder la topographie (Hestholm & Ruud, 1998; Hestholm, 1999), l'utilisation d'opérateurs en différences finies modifiés (de la Puente *et al.*, 2014), une approche hybride différences finies/éléments finis (Moczo *et al.*, 1997) et l'utilisation de la frontière immergée Gao *et al.* (2015); Hu (2016). Étant donné la complexité de simuler une surface libre incluant une topographie avec la méthode des différences finies, cette question reste un intense sujet de recherche jusqu'à ce jour.

Une méthode simple et largement populaire reste la méthode image proposée par Levander (1988). Nous présentons la version de Robertsson (1996), adaptée à l'équation viscoélastique et permettant la présence d'une topographie en escalier. Pour une surface libre horizontale à $z=0$, la condition de surface libre donne explicitement

$$\sigma_{xz} = \sigma_{yz} = \sigma_{zz} = 0, \quad (2.66a)$$

$$r_{xz}^l = r_{yz}^l = r_{zz}^l = 0. \quad (2.66b)$$

En insérant cette condition dans l'équation (2.29) pour les contraintes respectives, les conditions suivantes peuvent être obtenues:

$$\frac{\partial v_z}{\partial z} = - \left(1 - \frac{2\mu^0}{M^0} \right) \left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} \right), \quad (2.67a)$$

$$\frac{\partial v_x}{\partial z} = - \frac{\partial v_z}{\partial x}, \quad (2.67b)$$

$$\frac{\partial v_y}{\partial z} = - \frac{\partial v_z}{\partial y}. \quad (2.67c)$$

L'implémentation de la méthode image se fait donc comme suit. Les variables de mémoire aux noeuds non décalés en z sont fixées à 0, c'est-à-dire

$$\sigma_{zz}^{p,q,0} = 0, \quad (2.68a)$$

$$r_{zz}^{l\ p,q,0} = 0. \quad (2.68b)$$

Deuxièmement, les vitesses au-dessus de la surface libre sont fixées à 0 et les contraintes en z sont imagées de façon antisymétrique en prenant compte de la grille décalée:

$$\sigma_{zz}^{p,q,-n} = -\sigma_{zz}^{p,q,n+1}, \quad (2.69a)$$

$$\sigma_{xz}^{p,q,-n} = -\sigma_{xz}^{p,q,n}, \quad (2.69b)$$

$$\sigma_{yz}^{p,q,-n} = -\sigma_{yz}^{p,q,n}, \quad (2.69c)$$

$$r_{zz}^{l\ p,q,-n} = -r_{zz}^{l\ p,q,n+1}, \quad (2.69d)$$

$$r_{xz}^{l\ p,q,-n} = -r_{xz}^{l\ p,q,n}, \quad (2.69e)$$

$$r_{yz}^{l\ p,q,-n} = -r_{yz}^{l\ p,q,n}. \quad (2.69f)$$

Finalement, les contraintes et les variables de mémoire situées directement à la surface libre sont mises à jour en remplaçant les dérivées spatiales des vitesses en z par les relations de l'équa-

tion (2.67):

$$\begin{aligned} \sigma_{xx}^{p,q,0,t+1/2} = & \sigma_{xx}^{p,q,0,t-1/2} + \Delta t 2\mu^0{}^{p,q,0} \left(D_x^- v_x^{p,q,0,t} + \left(1 - \frac{2\mu_0^{p,q,0}}{M_0^{p,q,0}} \right) \left(D_x^- v_x^{p,q,0,t} + D_y^- v_y^{p,q,0,t} \right) \right) \\ & + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{xx}^l{}^{p,q,0,t} + r_{xx}^l{}^{p,q,0,t+1} \right), \end{aligned} \quad (2.70a)$$

$$\begin{aligned} \sigma_{yy}^{p,q,0,t+1/2} = & \sigma_{yy}^{p,q,0,t-1/2} + \Delta t 2\mu^0{}^{p,q,0} \left(D_y^- v_y^{p,q,0,t} + \left(1 - \frac{2\mu_0^{p,q,0}}{M_0^{p,q,0}} \right) \left(D_x^- v_x^{p,q,0,t} + D_y^- v_y^{p,q,0,t} \right) \right) \\ & + \frac{\Delta t}{2} \sum_{l=1}^L \left(r_{yy}^l{}^{p,q,0,t} + r_{yy}^l{}^{p,q,0,t+1} \right), \end{aligned} \quad (2.70b)$$

$$\begin{aligned} r_{xx}^l{}^{p,q,r,t+1} = & \left(1 - \frac{1}{2\tau_{\sigma l}} \right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} M^l{}^{p,q,r} \frac{2\mu^l{}^{p,q,0}}{M^0{}^{p,q,0}} \left(D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} \right) \right. \\ & - \frac{\Delta t}{\tau_{\sigma l}} 2\mu^l{}^{p,q,r} \left(D_y^- v_y^{p,q,r,t} - \left(1 - \frac{2\mu^0{}^{p,q,0}}{M^0{}^{p,q,0}} \right) \left(D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} \right) \right) \\ & \left. + \left(1 + \frac{1}{2\tau_{\sigma l}} \right) r_{xx}^l{}^{p,q,r,t} \right], \end{aligned} \quad (2.70c)$$

$$\begin{aligned} r_{yy}^l{}^{p,q,r,t+1} = & \left(1 - \frac{1}{2\tau_{\sigma l}} \right)^{-1} \left[\frac{\Delta t}{\tau_{\sigma l}} M^l{}^{p,q,r} \frac{2\mu^0{}^{p,q,0}}{M^0{}^{p,q,0}} \left(D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} \right) \right. \\ & - \frac{\Delta t}{\tau_{\sigma l}} 2\mu^l{}^{p,q,r} \left(D_x^- v_x^{p,q,r,t} - \left(1 - \frac{2\mu^0{}^{p,q,0}}{M^0{}^{p,q,0}} \right) \left(D_x^- v_x^{p,q,r,t} + D_y^- v_y^{p,q,r,t} \right) \right) \\ & \left. + \left(1 + \frac{1}{2\tau_{\sigma l}} \right) r_{yy}^l{}^{p,q,r,t} \right]. \end{aligned} \quad (2.70d)$$

2.3 Inversion

2.3.1 Formulation du problème

Comme mentionné précédemment, le but de l'inversion est de retrouver les paramètres \mathbf{m} à partir des données mesurées, réponse d'un phénomène physique $\mathbf{d} = \mathcal{F}(\mathbf{m})$. Pour l'inversion de formes d'ondes complètes viscoélastique isotrope, les paramètres physiques devant être reconstruits sont par exemple:

$$\mathbf{m} = [\mathbf{V}_p, \mathbf{V}_s, \rho, \mathbf{Q}_p, \mathbf{Q}_s]. \quad (2.71)$$

Ici, les vitesses des ondes de compression V_p et de cisaillement V_s sont utilisées à la place des modules des ondes P , M , et de cisaillement, μ . Ces paramètres sont toutefois reliés par

$$V_p = \sqrt{\frac{M}{\rho}}, \quad (2.72)$$

$$V_s = \sqrt{\frac{\mu}{\rho}}. \quad (2.73)$$

À noter que plusieurs types de paramétrisation sont possibles, et certaines combinaisons sont plus avantageuses que d'autres selon la configuration d'acquisition (Tarantola, 1986; Plessix & Cao, 2011; Operto *et al.*, 2013; Gholami *et al.*, 2013). Par simplicité, nous nous restreignons à la paramétrisation utilisant les vitesses sismiques dans ce travail. Cette reconstruction se base sur la mesure des vitesses de déplacement des particules du sous-sol échantillonnées à des points discrets dans l'espace et dans le temps:

$$\mathbf{d} = [\mathbf{S}v_x, \mathbf{S}v_y, \mathbf{S}v_z], \quad (2.74)$$

où \mathbf{S} est l'opérateur d'échantillonnage qui échantillonne la vitesse des particules aux temps et positions discrètes des receveurs. De telles mesures sont acquises par des géophones trois composantes lors de levés sismiques, par exemple. Le modèle direct, $\mathcal{F}(\mathbf{m})$, est l'équation d'onde générale (équation (2.24)), ou une de ses approximations, telle que l'équation d'onde viscoélastique et isotrope donnée à l'équation (2.29). Comme l'équation d'onde ne relie pas explicitement de façon linéaire la mesure des vitesses des particules aux paramètres sismiques, l'inversion se base sur la minimisation d'une fonction coût. Le fondement de cette approche est l'observation qu'en utilisant les paramètres exacts du modèle vrai, les données mesurées correspondent exactement aux données modélisées, et donc que la fonction coût atteint un minimum à ce point. En recherchant les paramètres qui minimisent la fonction coût, il est donc possible de retrouver les paramètres physiques du modèle. L'expression générale de la fonction coût utilisée dans ce travail est donnée par

$$\mathcal{X}(\mathbf{m}) = \|\mathfrak{F}[\mathbf{S}\mathcal{F}(\mathbf{m})] - \mathfrak{F}[\mathbf{d}]\|_p. \quad (2.75)$$

La fonction \mathfrak{F} exprime le fait que les données brutes peuvent être transformées par une fonction quelconque, par exemple par une transformée de Fourier. Communément, la norme l_2 ($p = 2$) est utilisée en inversion d'onde complète, en utilisant directement les données telles qu'enregistrées, ce

qui donne

$$\mathcal{X}(\mathbf{m}) = \frac{1}{2} (\mathbf{S}\mathcal{F}(\mathbf{m}) - \mathbf{d})^T (\mathbf{S}\mathcal{F}(\mathbf{m}) - \mathbf{d}). \quad (2.76)$$

La minimisation d'une telle fonction coût est le domaine de la minimisation des moindres carrées non contrainte. Modrak & Tromp (2016) offrent une bonne revue des différentes stratégies d'optimisation et de régularisation utilisées en inversion de formes d'ondes complètes. Les éléments essentiels seront repris dans ce qui suit.

2.3.2 Fondement des méthodes de recherche linéaire

Il est impossible de calculer globalement l'aspect de la fonction coût afin de trouver son minimum global en raison du coût important de la résolution numérique de l'équation d'onde. Il faut donc se résoudre à utiliser une information locale au modèle initial. Classiquement, deux familles de méthodes itératives sont utilisées pour trouver localement le minimum de la fonction coût: les méthodes de recherche linéaire et les méthodes des régions de confiance (Nocedal & Wright, 2006). Dans ce travail, nous nous concentrons uniquement sur les méthodes de recherche linéaire, qui cherchent à trouver une direction et une grandeur de pas afin de converger vers le minimum. Ces méthodes prennent la forme

$$\mathbf{m}^{k+1} = \mathbf{m}^k - \alpha^k \mathbf{p}^k. \quad (2.77)$$

Le paramètre α est la grandeur de pas et \mathbf{p}^k est la direction de descente. Un critère pour déterminer la direction de descente peut être obtenu en développant en série de Taylor la fonction coût autour de $\mathbf{m} + \alpha^k \mathbf{p}$:

$$\mathcal{X}(\mathbf{m} + \alpha^k \mathbf{p}) = \mathcal{X}(\mathbf{m}) + \alpha \mathbf{p}^T \nabla \mathcal{X}(\mathbf{m}) + \mathcal{O}(\alpha^2). \quad (2.78)$$

Ainsi, pour une grandeur de pas assez petite, la fonction coût décroît, $\mathcal{X}(\mathbf{m} + \alpha^k \mathbf{p}) < \mathcal{X}(\mathbf{m})$, si

$$\mathbf{p}^T \nabla \mathcal{X}(\mathbf{m}) < 0. \quad (2.79)$$

Cette dernière équation est la condition de descente que doit respecter la direction de recherche pour produire une réduction de la fonction coût. Cette condition suggère de choisir la direction de descente comme l'opposée du gradient de la fonction coût, c'est-à-dire

$$\mathbf{p} = -\nabla\mathcal{X}(\mathbf{m}). \quad (2.80)$$

Ce choix résulte en l'algorithme de la plus forte descente. Il est possible d'obtenir une direction différente en conservant les termes de second ordre dans le développement de la série de Taylor:

$$\mathcal{X}(\mathbf{m} + \mathbf{p}) = \mathcal{X}(\mathbf{m}) + \mathbf{p}^T \nabla\mathcal{X}(\mathbf{m}) + \frac{1}{2} \mathbf{p}^T \nabla^2 \mathcal{X}(\mathbf{m}) \mathbf{p} + \mathcal{O}(\mathbf{p}^3). \quad (2.81)$$

À son minimum, la dérivée de la fonction coût par rapport à une perturbation doit être nulle. En négligeant les termes d'ordre supérieurs à 2, nous avons

$$\nabla^2 \mathcal{X}(\mathbf{m}) \mathbf{p} = -\nabla\mathcal{X}(\mathbf{m}). \quad (2.82)$$

En supposant que $\nabla^2 \mathcal{X}$ est définie positive, ceci donne la formule d'optimisation suivante:

$$\mathbf{m}^{k+1} = \mathbf{m}^k - \nabla^2 \mathcal{X}^{k-1} \nabla \mathcal{X}^k, \quad (2.83)$$

avec $\nabla^2 \mathcal{X}^k = \nabla^2 \mathcal{X}(\mathbf{m}^k)$ et $\nabla \mathcal{X}^k = \nabla \mathcal{X}(\mathbf{m}^k)$. Cette dernière équation consiste en l'algorithme de Newton. Notons que si la seconde dérivée de la fonction coût n'est pas définie positive, ou si la direction $\nabla^2 \mathcal{X}^{-1} \nabla \mathcal{X}$ ne respecte pas la condition de descente, il est possible que la méthode de Newton n'arrive pas à converger. La régularisation est alors de mise (voir Tarantola (2005) sur ce point).

2.3.3 Dimensions du problème d'inversion

L'inversion de formes d'ondes complètes constitue un problème d'inversion de très grande taille. En effet, le nombre de données N_d ainsi que le nombre de paramètres d'inversion N_m peuvent facilement atteindre des tailles telles que certains algorithmes d'optimisation ne sont pas applicables étant données les capacités de calcul actuelles. Ceci peut être démontré en étudiant plus en détail

la structure du gradient $\nabla\mathcal{X}$ et de la Hessienne $\nabla^2\mathcal{X}$ de la fonction coût. Le gradient est donné par

$$\nabla\mathcal{X}_\alpha(\mathbf{m}) = J_{\beta\alpha}(\mathbf{m})(S_\beta\mathcal{F}(\mathbf{m}) - d_\beta). \quad (2.84)$$

Cette dernière expression implique la matrice jacobienne, dont les éléments sont définis comme

$$J_{\alpha\beta}(\mathbf{m}) = S_\alpha \frac{\partial\mathcal{F}(\mathbf{m})}{\partial m_\beta}. \quad (2.85)$$

Ainsi, cette matrice possède un nombre de colonnes égal au nombre des données N_d et un nombre de rangées égal au nombre de paramètres d'inversion N_m . La Hessienne est quant à elle donnée par

$$\nabla^2\mathcal{X}_{\alpha\beta}(\mathbf{m}) = J_{\gamma\alpha}(\mathbf{m})J_{\gamma\beta}(\mathbf{m}) + \frac{\partial J_{\gamma\alpha}(\mathbf{m})}{\partial m_\beta}(S_\gamma\mathcal{F}(\mathbf{m}) - d_\gamma). \quad (2.86)$$

La Hessienne est une matrice de $N_m \times N_m$ composantes, qui doit être inversée si la méthode de Newton est utilisée selon sa forme standard. En pratique, la matrice jacobienne et la Hessienne sont trop volumineuses pour être calculées explicitement. Par exemple, pour un levé avec 100 tirs, comportant 100 traces avec 5000 échantillons, le nombre de données est $N_d = 500 \times 10^6$. De plus, supposons une grille d'inversion 2D de 1000×1000 , c'est-à-dire $N_m = 10^6$, en n'inversant que pour un seul type de paramètre (V_p par exemple). Pour ce levé modeste, la matrice jacobienne contient 500×10^{12} éléments, ce qui demande des ressources en mémoire considérable. De plus, le calcul de cette matrice par différences finies demande $N_m + 1$ modélisations directes, ce qui est impensable considérant le coût élevé de la modélisation de l'équation d'onde. De la même façon, pour ce levé, la matrice Hessienne serait de dimension $10^6 \times 10^6$, qui devrait elle aussi être stockée en mémoire. L'inversion d'une matrice de telle taille est aussi grandement problématique avec les méthodes de calculs numériques existantes. Pour ces raisons, les algorithmes d'optimisation en inversion de formes d'ondes complètes se basent sur des méthodes ne requérant pas le calcul de la matrice jacobienne, telle la méthode adjointe qui donne directement le gradient de la fonction coût $\nabla\mathcal{X}$. Celle-ci est décrite au chapitre suivant. De même, la Hessienne n'est jamais calculée et inversée explicitement. Des algorithmes qui approximent la Hessienne de façon itérative sont plutôt utilisés, dont les plus communs sont présentés dans ce qui suit.

Algorithme 1 Méthode de la descente la plus forte

```

1: Définir  $\mathbf{m}_0$ 
2:  $k = 0$ 
3: while  $\mathcal{X}^k > tol$  do
4:    $\mathbf{p}^k = -\nabla\mathcal{X}(\mathbf{m}^k)$ 
5:   Calculer  $\alpha^k$  par recherche linéaire
6:    $\mathbf{m}^{k+1} = \mathbf{m}^k + \alpha^k\mathbf{p}^k$ 
7:    $\mathcal{X}^{k+1} = \mathcal{X}(\mathbf{m}^{k+1})$ 
8:    $k = k + 1$ 

```

2.3.4 Méthodes itératives de recherche linéaire

Méthode de la descente la plus forte

La plus simple des méthodes de recherche linéaire est la méthode de la descente la plus forte, exprimée par

$$\mathbf{m}^{k+1} = \mathbf{m}^k - \alpha^k \nabla \mathcal{X}. \quad (2.87)$$

Celle-ci ne requiert que le calcul du gradient de la fonction coût et la détermination de la grandeur de pas. Le gradient de la fonction coût est obtenu à l'aide de la méthode adjointe, qui ne requiert environ que deux à trois modélisations directes.

La simplicité de l'algorithme de la plus forte descente a cependant un inconvénient majeur: une convergence très lente, voire une absence de convergence (Pratt *et al.*, 1998). Comme un très grand nombre d'itérations est requis, ceci peut être coûteux en temps de calcul. Malgré cet inconvénient, la méthode de la descente la plus forte est souvent utilisée (Gauthier *et al.*, 1986; Sambridge *et al.*, 1991), de façon plus commune avec régularisation (Shin *et al.*, 2001; Ravaut *et al.*, 2004; Operto *et al.*, 2006).

Méthode du gradient conjugué non linéaire

La méthode du gradient conjugué non linéaire vise à minimiser l'approximation en série de Taylor du second ordre de la fonction coût:

$$\mathcal{X}(\mathbf{m} + \alpha\mathbf{p}) = \mathcal{X}(\mathbf{m}) + \alpha\mathbf{p}^T \nabla \mathcal{X}(\mathbf{m}) + \alpha^2 \mathbf{p}^T \nabla^2 \mathcal{X}(\mathbf{m}) \mathbf{p}. \quad (2.88)$$

La fonction coût est supposée purement quadratique pour l'instant, ce qui permet de dire que $\nabla^2 \mathcal{X}(\mathbf{m} + \mathbf{p}) = \nabla^2 \mathcal{X}(\mathbf{m}) = \nabla^2 \mathcal{X}$ peu importe les paramètres.

La méthode du gradient conjugué, tout comme la méthode de la descente la plus forte, se base sur le gradient de la fonction coût. Cependant, une contrainte supplémentaire est introduite: les directions de descentes doivent être conjuguées. Deux vecteurs sont conjugués si

$$\mathbf{p}^i \nabla^2 \mathcal{X} \mathbf{p}^j = 0, \quad \text{pour } i \neq j. \quad (2.89)$$

Une direction de descente conjuguée peut être obtenue par la relation de récurrence suivante:

$$\mathbf{p}^k = -\nabla \mathcal{X}^k + \beta^k \mathbf{p}^{k-1}. \quad (2.90)$$

La constante β est obtenue en prémultipliant cette expression par $\mathbf{p}^{k-1T} \nabla^2 \mathcal{X}$ et en utilisant la définition de la conjugaison, ce qui donne

$$\beta^k = \frac{\mathbf{p}^{k-1T} \nabla^2 \mathcal{X} \nabla \mathcal{X}^k}{\mathbf{p}^{k-1T} \nabla^2 \mathcal{X} \mathbf{p}^{k-1}}. \quad (2.91)$$

Une relation alternative est obtenue en calculant le gradient de la fonction coût à un point quelconque $\nabla \mathcal{X}^{k+1} = \nabla \mathcal{X}(\mathbf{m} + \alpha^k \mathbf{p}^k)$:

$$\nabla \mathcal{X}^{k+1} = \nabla \mathcal{X}^k + \alpha^k \nabla^2 \mathcal{X} \mathbf{p}^k, \quad (2.92)$$

ce qui permet de réécrire l'expression pour β^k comme

$$\beta_{FR} = \frac{\nabla \mathcal{X}^k \nabla \mathcal{X}^k}{\nabla \mathcal{X}^{k-1T} \nabla \mathcal{X}^{k-1}}. \quad (2.93)$$

Cette forme du paramètre conjugué mène à l'algorithme de Fletcher-Reeves. Comme pour un cas purement quadratique, les directions des gradients sont conjuguées, il est aussi possible de réécrire le paramètre conjugué comme

$$\beta_{PR} = \frac{\nabla \mathcal{X}^k T (\nabla \mathcal{X}^k - \nabla \mathcal{X}^{k-1})}{\nabla \mathcal{X}^{k-1} T \nabla \mathcal{X}^{k-1}}. \quad (2.94)$$

Cette dernière forme consiste en l'algorithme de Polak-Ribière. Dans le cas purement quadratique, ces deux algorithmes sont identiques. Cependant, dans le cas général où la fonction coût ne peut plus être exactement décrite par une série de Taylor du second ordre, les deux algorithmes sont différents. Dans ce cas, la formulation de Polak-Ribière est avantageuse, car plus robuste et efficace (Nocedal & Wright, 2006). Un autre effet de la non-linéarité de la fonction coût est qu'à un certain moment, les directions conjuguées ne correspondront plus à des directions de descente. Il faut alors réinitialiser l'algorithme, ce qui peut être fait automatique à l'aide du critère suivant:

$$\beta = \max \{ \beta_{PR}, 0 \}. \quad (2.95)$$

La méthode du gradient conjugué de Polak-Ribière est résumée à l'Algorithme 2. Le gradient conjugué est fréquemment utilisé en inversion de formes d'ondes complètes, car son coût est équivalent à la méthode de la descente la plus forte, tout en ayant une meilleure convergence (voir par exemple Shipp & Singh (2002); Mora (1987)).

Algorithme 2 Méthode du gradient conjugué non linéaire

- 1: **Calculer** $\mathcal{X}^0 = \mathcal{X}(\mathbf{m}^0)$, $\mathbf{r}^0 = \nabla \mathcal{X}(\mathbf{m}^0)$
 - 2: **Fixer** $p^0 = -\mathbf{r}^0$, $k \leftarrow 0$
 - 3: **while** $\frac{\mathcal{X}^k - \mathcal{X}^{k-1}}{\mathcal{X}^k} > tol$ **do**
 - 4: **Obtenir** α^k par recherche linéaire
 - 5: $\mathbf{m}^{k+1} = \mathbf{m}^k + \alpha^k \mathbf{p}^k$
 - 6: **Calculer** $\mathbf{r}^{k+1} = -\nabla \mathcal{X}(\mathbf{m}^{k+1})$
 - 7: $\beta_{PR}^k = \frac{\mathbf{r}^{kT}(\mathbf{r}^k - \mathbf{r}^{k-1})}{\mathbf{r}^{k-1T} \mathbf{r}^{k-1}}$
 - 8: $\beta = \max \{ \beta_{PR}, 0 \}$
 - 9: $\mathbf{p}^{k+1} = \mathbf{r}^{k+1} + \beta^k \mathbf{p}^k$
 - 10: $\mathcal{X}^{k+1} = \mathcal{X}(\mathbf{m}^{k+1})$
 - 11: $k = k + 1$
-

Méthode de Newton tronquée

Contrairement à la méthode du gradient conjugué non linéaire, la méthode de Newton tronquée ne suppose pas une Hessienne constante au voisinage de la solution. Ainsi, à chaque itération, le système de l'équation (2.82) doit être résolu:

$$\nabla^2 \mathcal{X}^k \mathbf{p}^k = -\nabla \mathcal{X}^k. \quad (2.96)$$

Selon la méthode de Newton, la Hessienne doit être explicitement construite et inversée, ce qui n'est pas applicable pour l'inversion de formes d'ondes complètes étant donné la grande taille du problème. La méthode de Newton tronquée contourne ce problème en utilisant un algorithme itératif tel le gradient conjugué linéaire pour résoudre ce système linéaire. Dans ce cas, il n'est pas requis de construire la Hessienne au complet, seul le calcul du produit d'une rangée de la Hessienne et du vecteur de direction suffit. Ceci peut être calculé par différences finies au coût d'une seule modélisation par itération du gradient conjugué. Théoriquement, l'algorithme du gradient conjugué nécessite au plus N_m itérations pour résoudre un système linéaire. En pratique, il peut être arrêté bien avant, d'où le terme tronqué de la méthode. L'arrêt des itérations peut être décidé lorsque le changement de direction de la direction de recherche devient plus faible qu'un certain seuil.

L'algorithme de Newton tronqué est résumé à l'Algorithme 3. Cet algorithme contient deux boucles itératives: une boucle externe qui met à jour le vecteur des paramètres et une boucle interne qui corrige la direction du vecteur de correction par gradient conjugué. Le nombre de modélisations directes supplémentaires requis par la boucle interne en comparaison aux algorithmes de gradient conjugué non linéaire et à la méthode de plus forte descente est compensé par une meilleure convergence, moins susceptible d'être pris dans des minimums locaux (voir Métivier *et al.* (2013) pour le développement de cette méthode dans le contexte de l'inversion en complète).

Méthodes de Quasi-Newton

Les méthodes de Quasi-Newton se basent aussi sur un développement en série de Taylor d'ordre 2 de la fonction coût:

$$\mathcal{X}(\mathbf{m} + \alpha \mathbf{p}) = \mathcal{X}(\mathbf{m}) + \alpha \mathbf{p}^T \nabla \mathcal{X}(\mathbf{m}) + \alpha^2 \mathbf{p}^T \mathbf{B} \mathbf{p}. \quad (2.97)$$

Algorithme 3 Méthode de Newton tronquée

```

1: Définir  $\mathbf{m}_0$ 
2: Calculer  $\mathcal{X}_0 = \mathcal{X}(\mathbf{m}_0)$ ,  $\nabla \mathcal{X}_0 = \nabla \mathcal{X}(\mathbf{m}_0)$ 
3: Définir  $A_k(\mathbf{b}) = \frac{\nabla \mathcal{X}(\mathbf{m}_k + \epsilon \mathbf{b}) - \nabla \mathcal{X}(\mathbf{m}_k)}{\epsilon}$ 
4:  $\mathbf{p}_0^0 = -\nabla \mathcal{X}_0$ 
5:  $k = 0$ 
6: while  $\mathcal{X}_k > tol$  do
7:    $\mathbf{r}_k^0 = A_k(\mathbf{p}_k^0) + \nabla \mathcal{X}_k$ 
8:    $\mathbf{q}^0 = -\mathbf{r}_0$ 
9:    $n = 0$ 
10:  while  $n < n_{max}$  do
11:     $\alpha_k^n = \frac{\mathbf{r}^{nT} \mathbf{r}^n}{\mathbf{q}^{nT} A_k(\mathbf{q}^n)}$ 
12:     $\mathbf{p}_k^{n+1} = \mathbf{p}_k^n + \alpha_k^n \mathbf{q}^n$ 
13:     $\mathbf{r}_k^{n+1} = \mathbf{r}_k^n + \alpha_k^n A_k(\mathbf{q}^n)$ 
14:     $\beta^{n+1} = \frac{\mathbf{r}^{n+1T} \mathbf{r}^{n+1}}{\mathbf{r}^{nT} \mathbf{r}^n}$ 
15:     $\mathbf{q}_k^{n+1} = -\mathbf{r}_k^{n+1} + \beta^{n+1} \mathbf{q}^n$ 
16:     $n = n + 1$ 
17:  Calculer  $\alpha_k$  par recherche linéaire avec  $\alpha_k = 1$ 
18:   $\mathbf{m}_{k+1} = \mathbf{m}_k + \alpha_k \mathbf{p}_k^n$ 
19:  Calculer  $\mathcal{X}_{k+1} = \mathcal{X}(\mathbf{m}_{k+1})$ ,  $\nabla \mathcal{X}_{k+1} = \nabla \mathcal{X}(\mathbf{m}_{k+1})$ 
20:   $\mathbf{p}_{k+1}^0 = -\nabla \mathcal{X}_{k+1}$ 
21:   $k = k + 1$ 

```

Ici, la vraie Hessienne, $\nabla^2 \mathcal{X}$, a été remplacée par une approximation \mathbf{B} . La direction de recherche est alors donnée par

$$\mathbf{p}^k = -\mathbf{H}^k \nabla \mathcal{X}^k, \quad (2.98)$$

où l'inverse de la Hessienne est dénoté $\mathbf{H}^k = \mathbf{B}^{k-1}$. La méthode développée par Boyd, Fletcher, Goldfarb et Shanno, communément appelée BFGS, approxime l'inverse de la Hessienne en utilisant la courbure mesurée entre les différentes itérations, selon la formule

$$\mathbf{H}^{k+1} = \left(\mathbf{I} - \frac{\mathbf{s}^k \mathbf{y}^{kT}}{\mathbf{s}^{kT} \mathbf{y}^k} \right) \mathbf{H}^k \left(\mathbf{I} - \frac{\mathbf{y}^k \mathbf{s}^{kT}}{\mathbf{s}^{kT} \mathbf{y}^k} \right) + \frac{\mathbf{s}^k \mathbf{s}^{kT}}{\mathbf{s}^{kT} \mathbf{y}^k}, \quad (2.99)$$

où $\mathbf{s}^k = \mathbf{m}^{k+1} - \mathbf{m}^k$ et $\mathbf{y}^k = \nabla \mathcal{X}^{k+1} - \nabla \mathcal{X}^k$. Ceci requiert de garder en mémoire une matrice de taille $N_m \times N_m$. Pour pallier à ce fait, les méthodes à mémoire limitée l-BFGS ont été proposées. Celles-ci proposent de ne conserver en mémoire que les m paires de vecteurs $\{\mathbf{s}^i, \mathbf{y}^i\}$. L'approximation de \mathbf{H}^k peut ensuite être retrouvée à l'aide d'une formule de récursion telle que présentée à l'Algorithme

Algorithme 4 Méthode de l-BFGS

```

1: Définir  $\mathbf{m}^0$ 
2: Calculer  $\mathcal{X}^0 = \mathcal{X}(\mathbf{m}^0)$ ,  $\nabla \mathcal{X}^0 = \nabla \mathcal{X}(\mathbf{m}^0)$ 
3:  $\mathbf{p}^0 = \nabla \mathcal{X}^0$ 
4:  $k = 0$ 
5: while  $\mathcal{X}^k > tol$  do
6:   for  $i = k - 1, k - 2, \dots, k - m$  do
7:      $\gamma^i = (\mathbf{y}^{iT} \mathbf{s}^i)^{-1} \mathbf{s}^{iT} \mathbf{p}^k$ 
8:      $\mathbf{p}^k = \mathbf{p}^k - \gamma^i \mathbf{y}^i$ 
9:      $\mathbf{p}^k = (\mathbf{y}^{k-1T} \mathbf{y}^{k-1})^{-1} (\mathbf{s}^{k-1T} \mathbf{y}^{k-1}) \mathbf{p}^k$ 
10:   for  $i = k - m, k - m + 1, \dots, k - 1$  do
11:      $\lambda = (\mathbf{y}^{iT} \mathbf{s}^i)^{-1} \mathbf{y}^{iT} \mathbf{p}^k$ 
12:      $\mathbf{p}^k = \mathbf{p}^k + \mathbf{s}^i (\gamma^i - \lambda)$ 
13:    $\mathbf{p}^k = -\mathbf{p}^k$ 
14:   Calculer  $\alpha^k$  par recherche linéaire avec  $\alpha^k = 1$ 
15:    $\mathbf{m}^{k+1} = \mathbf{m}^k + \alpha^k \mathbf{p}^k$ 
16:   Calculer  $\mathcal{X}^{k+1} = \mathcal{X}(\mathbf{m}^{k+1})$ ,  $\nabla \mathcal{X}^{k+1} = \nabla \mathcal{X}(\mathbf{m}^{k+1})$ 
17:    $\mathbf{p}^{k+1} = \nabla \mathcal{X}^{k+1}$ 
18:   Sauvegarder  $\mathbf{s}^k = \mathbf{m}^{k+1} - \mathbf{m}^k$  et  $\mathbf{y}^k = \nabla \mathcal{X}^{k+1} - \nabla \mathcal{X}^k$ 
19:   if  $k > m$  then
20:     Discarter  $\mathbf{s}^{k-m}$  et  $\mathbf{y}^{k-m}$ 
20:    $k = k + 1$ 

```

4. Notons qu'étant donné son faible coût et une nette amélioration de la convergence, l'algorithme l-BFGS est abondamment utilisé en inversion de formes d'ondes complètes (Brossier *et al.*, 2009; Prioux *et al.*, 2013; Asnaashari *et al.*, 2013).

Descente par gradient stochastique

L'inversion par gradient stochastique (SGD) est une approche communément utilisée dans la communauté de la *Big Data* pour les problèmes de très grande taille (Bottou, 2010). Comme pour l'algorithme de la descente la plus forte, le SGD utilise le gradient de la fonction coût comme direction de descente. Cependant, le SGD utilise un sous-ensemble aléatoire des données pour calculer le gradient, qui change à chacune des itérations. Formellement, la formule de mise à jour est donnée par

$$\mathbf{m}^{k+1} = \mathbf{m}^k - \alpha \nabla \mathcal{X}_\Omega(\mathbf{m}^k), \quad (2.100)$$

Algorithme 5 Méthode du gradient stochastique

```

1: Définir  $\mathbf{m}^0$ 
2:  $k = 0$ 
3: while  $\mathcal{X}^k > tol$  do
4:   Choisir aléatoirement  $\Omega(\mathbf{d})$ 
5:    $\mathbf{p}^k = -\nabla \mathcal{X}_\Omega(\mathbf{m}^k)$ 
6:   Calculer  $\alpha^k$  par recherche linéaire
7:    $\mathbf{m}^{k+1} = \mathbf{m}^k + \alpha^k \mathbf{p}^k$ 
8:    $\mathcal{X}^{k+1} = \mathcal{X}(\mathbf{m}^{k+1})$ 
9:    $k = k + 1$ 

```

où $\nabla \mathcal{X}_\Omega(\mathbf{m}^k)$ est le gradient calculé pour le sous-ensemble des données Ω . L'algorithme est explicité à l'algorithme 5. Le sous-ensemble est construit par une sélection de points de tirs. En effet, le coût du calcul du gradient est proportionnel au nombre de tirs à modéliser. Ainsi, la réduction du nombre de tirs entraîne directement une réduction du temps de calcul du gradient. Ceci affecte évidemment la qualité de la direction de descente. Or les données sismiques étant hautement redondantes, le sous-échantillonnage des sources n'affecte pas trop la convergence de l'algorithme. L'inversion par gradient stochastique a été appliquée à l'inversion de formes d'ondes complètes par van Leeuwen *et al.* (2011); van Leeuwen & Herrmann (2013), et par Li *et al.* (2012) avec régularisation l_1 . À noter que la correction de la Hessienne peut aussi être utilisée avec une approche stochastique (Schraudolph *et al.*, 2007; Martens, 2010; Zeiler, 2012)

Détermination de la grandeur de pas

Pour tous ces algorithmes itératifs de recherche linéaire, une grandeur de pas α doit être déterminée. Ce paramètre ne doit pas être trop élevé, sinon il n'est plus possible de négliger les termes de second ordre dans la série de Taylor de la fonction coût. De même, une longueur de pas trop courte augmentera le nombre d'itérations pour arriver au minimum. En général, la longueur de pas optimal donnant la plus forte décroissance de la fonction coût n'est pas nécessaire, une approximation suffit. La condition d'Armijo permet de juger si la grandeur de pas est adéquate. Cette condition est définie par

$$\mathcal{X}^{k+1} \leq \mathcal{X}^k + c_1 \alpha \nabla \mathcal{X}^k{}^T \mathbf{p}. \quad (2.101)$$

pour une constante c_1 entre 0 et 1, souvent égale à 10^{-4} . Cette condition permet d'éliminer les grandeurs de pas trop grandes, mais ne permet pas de juger si le pas est suffisamment grand. Pour ce faire, une deuxième condition est introduite, la condition de courbure exprimée par

$$\nabla \mathcal{X}^{k+1^T} \mathbf{p} \geq c_2 \nabla \mathcal{X}^{k^T} \mathbf{p}. \quad (2.102)$$

La constante c_2 est bornée entre 0 et 1. Une valeur typique pour celle-ci est de 0.9 pour les méthodes de Newton et quasi-Newton et de 0.1 pour le gradient conjugué non linéaire. Les conditions données aux équations (2.101) et (2.102) sont nommées les conditions de Wolf. Ces conditions suggèrent un algorithme simple pour déterminer la grandeur de pas: l'algorithme de la recherche linéaire à rebours (voir Algorithme 6). Ce dernier consiste à choisir un pas initial de grande taille et à le réduire jusqu'à ce que la diminution de la fonction coût respecte le critère d'Armijo. Notons que différentes stratégies existent en inversion de formes d'ondes complètes, en particulier la méthode de la parabole (Vigh *et al.*, 2009).

Algorithme 6 Recherche linéaire à rebours

- 1: Choisir $\alpha > 0$, η , $c_1 \in (0, 1)$
 - 2: **while** $\mathcal{X}(\mathbf{m} + \alpha \mathbf{m}) > \mathcal{X}(\mathbf{m}) + c_1 \alpha \nabla \mathcal{X}^T \mathbf{p}$ **do**
 - 3: $\alpha = \eta \alpha$
-

2.3.5 Le défi des minimums locaux

Idéalement, nous désirons trouver la solution globale de la fonction coût, c'est-à-dire le modèle qui, parmi tous les modèles possibles, possède la plus faible valeur de la fonction coût. Cependant, les précédents algorithmes d'optimisation sont des algorithmes locaux, c'est-à-dire qu'ils convergent vers un point stationnaire dans l'entourage du modèle de départ. Un point stationnaire, c'est-à-dire un point pour lequel le gradient de la fonction coût est nul, peut être un minimum local ou un point de sel. Dans les deux cas, ces points peuvent être assez éloignés du minimum global, produisant un modèle erroné. Dans certains cas, comme pour les fonctions convexes, tout point stationnaire correspond au minimum global de la fonction, et les méthodes locales sont très performantes.

Ceci n'est cependant pas vrai pour des fonctions plus complexes, notamment pour l'inversion de formes d'ondes complètes, qui est connue pour être hautement non linéaire. La cause de cette non-

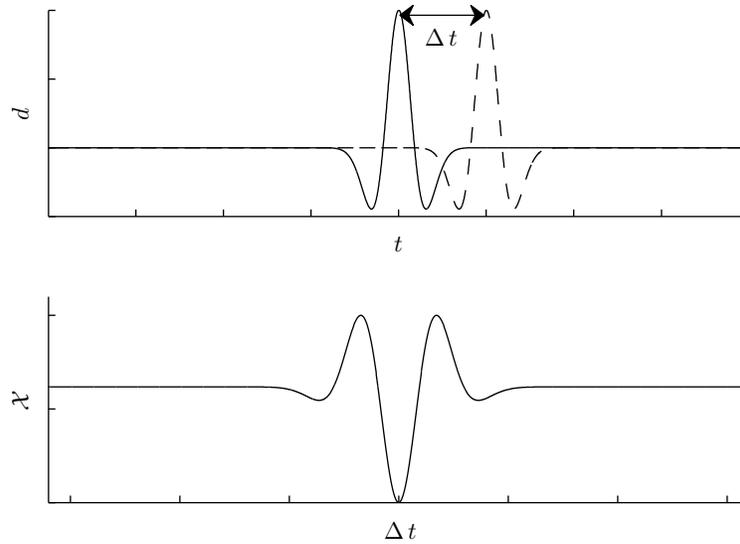


Figure 2.9 – Ondelette de référence (ligne continue) et ondelette modélisée (ligne pointillée) à la figure du haut. Lorsque la fonction coût est calculée en fonction de la distance Δt entre les deux ondelettes, la figure du bas est obtenue.

linéarité provient de deux sources : 1—l'équation d'onde en elle-même est hyperbolique et entraîne donc des phénomènes non linéaires, et 2—le phénomène de saut de cycle (Virieux & Operto, 2009). Le phénomène de saut de cycle survient lorsque deux fonctions ondulatoires séparées par plus de la moitié de la période de la phase dominante sont comparées. Comme le montre la Figure 2.9, la forme de la fonction coût basée sur la norme l_2 de la différence entre les traces présente la même forme que l'ondelette du signal. Notamment, deux minimum locaux sont observés de part et d'autre du minimum global. Dans ce cas, les algorithmes d'optimisation locale ne convergeront pas au minimum global si le modèle initial ne se trouve pas à plus ou moins la moitié de la période fondamentale de l'ondelette.

Avec les algorithmes d'inversion locale, il est donc nécessaire d'avoir un modèle initial assez proche du modèle optimal afin que le minimum local le plus proche corresponde au minimum global. Pour obtenir ce modèle, plusieurs méthodes sont possibles. Les approches basées sur les temps d'arrivées des réflexions sont couramment utilisées (voir Woodward *et al.* (2008) pour une revue). La tomographie des premières arrivées est aussi courante Zhang & Toksöz (1998). La stéréotomographie (Lambaré, 2008) est une approche plus récente qui utilise ces deux types d'arrivées et une méthode de pointé des arrivées semi-automatique.

Les méthodes mentionnées précédemment ne permettent cependant pas d'obtenir des modèles initiaux assez précis, c'est-à-dire des modèles qui ne sont pas affectés par le saut de cycle, et la

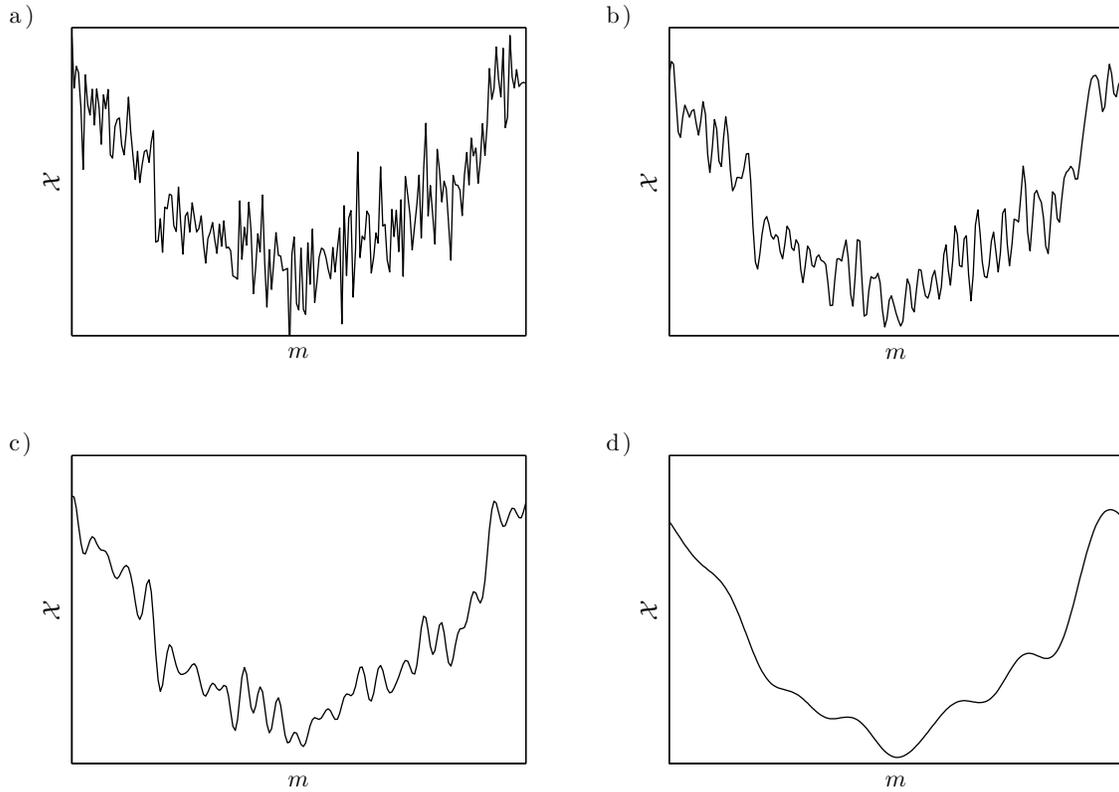


Figure 2.10 – Fonction coût pour des fréquences décroissantes de a) vers d).

régularisation de l'inversion de formes d'ondes complètes est nécessaire. La plupart des auteurs adoptent l'approche hiérarchique par fréquence (Bunks *et al.*, 1995; Sirgue & Pratt, 2004). L'inversion est effectuée fréquence par fréquence, des basses fréquences aux fréquences les plus élevées. Ceci permet d'éviter le plus possible l'effet du saut de cycle: les basses fréquences, de plus grande période, possèdent moins de minimums locaux. Comme le montre la Figure 2.10, plus la fréquence de l'ondelette est basse, plus la fonction coût est lisse et le nombre de minimums locaux diminue. La sophistication de cette méthode a été proposée par plusieurs auteurs, dont notamment Brossier *et al.* (2009) qui proposent une inversion simultanée selon plusieurs bandes de fréquences.

Une régularisation dans le domaine du temps est aussi possible. Par exemple, Sheng *et al.* (2006) débutent l'inversion par les temps courts, ce qui permet de n'utiliser que les premières arrivées au départ. L'inversion des premières arrivées contient moins de minimums locaux, ce qui permet de mieux converger vers le minimum global. Au cours de l'inversion, des temps de plus en plus longs sont utilisés ce qui permet d'inverser pour les arrivées plus tardives. L'inversion dans le domaine de Laplace (Shin & Cha, 2008; Shin & Ha, 2008) est une approche attrayante qui automatise ce processus.

Le problème de la non-convexité de la fonction coût est un intense sujet de recherche actuellement. Une tendance particulièrement intéressante est l'utilisation de fonctions coûts alternatives ayant une meilleure convexité que la comparaison conventionnelle entre traces brutes. Notons notamment la comparaison entre l'enveloppe et la phase des traces sismiques (Bozdağ *et al.*, 2011), l'approche par déformation dynamique des traces (Hale, 2013; Ma & Hale, 2013), la théorie du transport optimal (Métivier *et al.*, 2016), la comparaison des phases déroulées (Alkhalifah & Choi, 2014, 2012). Jiménez Tejero *et al.* (2015) présentent une comparaison de plusieurs de ces fonctions coûts. Quoique les résultats de ces différentes approches sont encourageants, aucune n'a jusqu'à maintenant apporté une réponse définitive à la question du saut de cycle.

Chapitre 3

Méthode adjointe

Toutes les méthodes de recherche linéaire présentées au chapitre précédent nécessitent le calcul du gradient de la fonction coût. Généralement, en inversion, celui-ci est obtenu en calculant la matrice jacobienne et en l'appliquant aux résidus (équation (2.84)). Cependant, la matrice jacobienne est trop volumineuse en pratique pour être contenue en mémoire. De plus, son calcul, lorsqu'il est effectué par différences finies, nécessite N_m modélisations directes. Il va sans dire qu'une approche différente est nécessaire. Cette méthode, qui est véritablement à la base de l'inversion de formes d'ondes complètes, se nomme la méthode adjointe. De N_m modélisations directes avec la matrice jacobienne, l'approche adjointe réduit ce coût à l'équivalent de 2 à 4 modélisations directes, peu importe le nombre de paramètres. Ceci est une réduction substantielle, et l'inversion en ondes complètes ne serait pas possible sans cette simplification.

La méthode adjointe a été proposée par Chavent (1974) dans un cadre général, c'est-à-dire à la minimisation d'une fonction coût impliquant des équations aux dérivées partielles. Lailly (1983) a ensuite appliquée la théorie à l'équation d'onde acoustique, ce qui constitue la première formulation de l'inversion en ondes complètes. Peu de temps après, Tarantola (1984a,b) a publié ce qui est considéré comme le fondement de l'inversion en ondes complètes. Tous deux utilisent l'opérateur adjoint de l'équation d'onde et la méthode adjointe à leur façon. Depuis, la méthode adjointe a été appliquée à plusieurs formulations de l'équation d'onde. Ainsi, Tarantola (1986) a généralisé son approche à l'équation d'onde élastique, puis à l'équation d'onde générale anisotrope viscoélastique (Tarantola, 1988). Tromp *et al.* (2004) reprend l'approche de Tarantola pour l'équation d'onde anisotrope viscoélastique dans une formulation plus moderne. Pratt (1990, 1999) a proposé l'approche

populaire en fréquence pour l'équation acoustique et élastique, étendue par après à l'équation d'onde viscoélastique (Song *et al.*, 1995; Liao & McMechan, 1995, 1996).

La formulation des équations adjointes dépend directement des équations aux dérivées partielles utilisées. Ainsi, ces équations auront une forme différente pour la formulation en déplacement ou pour la formulation en vitesse-contrainte, de même que pour l'équation d'onde dans le domaine fréquentiel ou temporel. Pour la formulation en vitesse-contrainte dans le domaine temporel de l'équation d'onde viscoélastique, la forme d'intérêt dans ce travail, la littérature est plutôt ténue. Fichtner & van Driel (2014) posent sommairement les équations dans cette formulation, mais leur dérivation est basée sur la formulation en déplacement et est peu explicitée. Récemment, Yang *et al.* (2016b) ont présenté une dérivation complète. Cependant, la formulation des équations adjointes diffère grandement des équations directes, ce qui requiert deux algorithmes différents de résolution numérique: un pour le modèle direct, un pour les équations adjointes.

Dans ce chapitre, une dérivation complète des équations adjointes de l'équation d'onde viscoélastique anisotrope pour le formalisme vitesse-contrainte est présentée. Le système d'équations aux dérivées partielles donnant les variables adjointes est identique à celui du modèle direct, à un signe près. Ceci permet d'utiliser pratiquement le même algorithme pour la modélisation directe et adjointe. Ces résultats font l'objet d'une publication soumise à la revue *Geophysics*:

- Fabien-Ouellet, G., Gloaguen, E. and Giroux, B., (2016) Anisotropic velocity-stress full waveform inversion. Soumis à *Geophysics*.

3.1 Fondements de la méthode

La méthode adjointe prend son fondement dans la méthode des multiplicateurs de Lagrange, utilisée en optimisation sous contraintes. La méthode des multiplicateurs de Lagrange résout le problème suivant, exprimé pour une contrainte unique:

$$\begin{aligned} & \underset{x}{\text{minimiser}} && f(\mathbf{x}), \\ & \text{sujet à} && g(\mathbf{x}) = 0. \end{aligned}$$

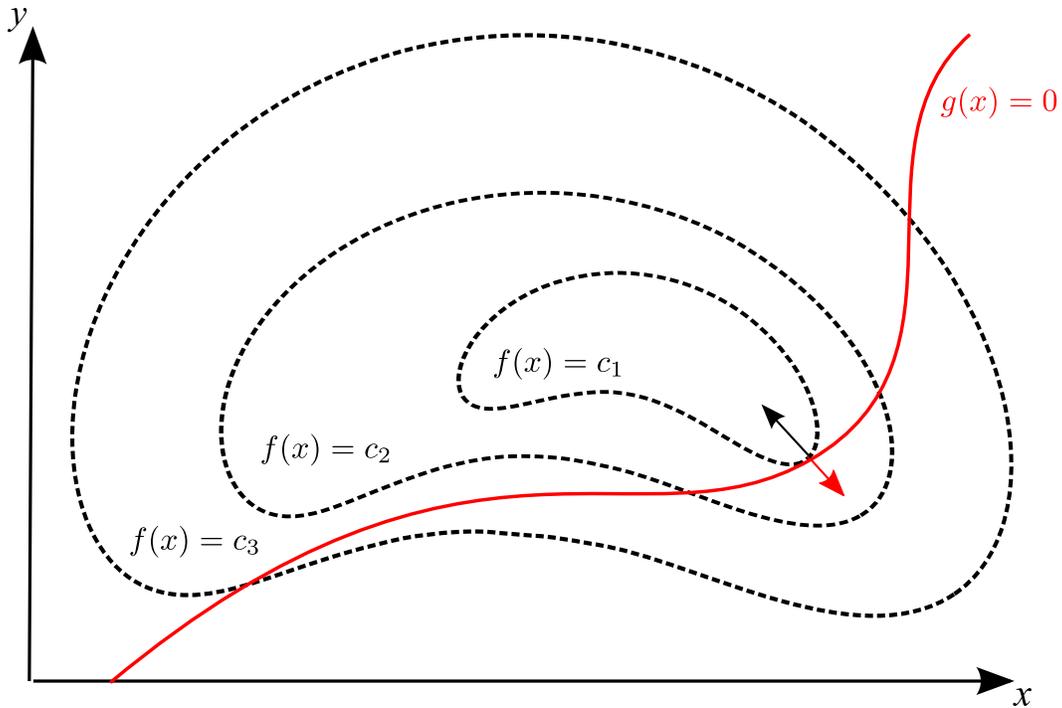


Figure 3.1 – Schéma des contours $f(x)$ d'une fonction à minimiser le long d'une contrainte $g(x)$.

Ce système peut être résolu en observant que le gradient de $f(\mathbf{x})$ doit être parallèle au gradient de la contrainte $g(\mathbf{x})$. En effet, lorsque la fonction $f(\mathbf{x})$ est minimum, l'isosurface minimale et la surface de la contrainte doivent être parallèles, sinon il serait possible de se déplacer le long de la contrainte pour obtenir une valeur plus faible de la fonction. Ce parallélisme, illustré à la figure 3.1, implique que les gradients sont parallèles.

Mathématiquement, ceci s'exprime comme

$$\nabla_X f(\mathbf{x}) = \lambda \nabla_X g(\mathbf{x}). \quad (3.1)$$

Il est donc possible de construire une fonction dont le gradient sera nul pour une valeur de λ respectant cette relation. Cette fonction se nomme le Lagrangien:

$$\mathcal{L} = f(x) - \lambda g(x). \quad (3.2)$$

La variable λ quant à elle se nomme le multiplicateur de Lagrange, une constante pour une seule contrainte, un vecteur pour un nombre plus élevé. En prenant la dérivée totale du Lagrangien et en l'égalant à 0, $N + 1$ équations sont obtenues, N étant le nombre de dimensions du problème

d'optimisation:

$$\nabla \mathcal{L} = \begin{bmatrix} g(\mathbf{x}) \\ \nabla_X f(\mathbf{x}) - \lambda \nabla_X g(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \quad (3.3)$$

La première égalité exprime la contrainte de départ. Le système de $N + 1$ équations peut alors être résolu pour trouver un point stationnaire, notamment le minimum de la fonction.

La méthode adjointe se base sur la généralisation de cette méthode à des fonctions continues et à un nombre de contraintes arbitraire (voir Plessix (2006) pour une excellente revue sur le sujet). La fonction à minimiser est bien entendu la fonction coût $\mathcal{X}(\boldsymbol{\phi}, \mathbf{m})$. La contrainte, quant à elle, correspond au modèle direct, exprimé comme $\mathcal{F}(\boldsymbol{\phi}, \mathbf{m}) = 0$. Ici, la variable d'état $\boldsymbol{\phi}$ correspond aux variables dynamiques du système d'équations aux dérivées partielles (par exemple les vitesses des particules et contraintes), tandis que \mathbf{m} correspond aux paramètres d'inversion (par exemple la densité, la vitesse sismique, le facteur de qualité). Le Lagrangien prend alors la forme suivante:

$$\mathcal{L}(\tilde{\boldsymbol{\phi}}, \tilde{\boldsymbol{\psi}}, \mathbf{m}) = \mathcal{X}(\tilde{\boldsymbol{\phi}}, \mathbf{m}) - \langle \tilde{\boldsymbol{\psi}}, \mathcal{F}(\tilde{\boldsymbol{\phi}}, \mathbf{m}) \rangle, \quad (3.4)$$

où $\tilde{\boldsymbol{\phi}}$ et $\tilde{\boldsymbol{\psi}}$ sont respectivement n'importe quelle réalisation de la variable d'état et de la variable adjointe. Les états respectant les contraintes sont obtenus en égalant la dérivée totale du Lagrangien à 0:

$$\nabla \mathcal{L} = \begin{bmatrix} \mathcal{F}(\boldsymbol{\phi}, \mathbf{m}) \\ \nabla_{\boldsymbol{\phi}} \mathcal{X}(\boldsymbol{\phi}, \mathbf{m}) - \mathcal{F}^*(\boldsymbol{\psi}, \mathbf{m}) \\ \nabla_M \mathcal{X}(\boldsymbol{\phi}, \mathbf{m}) - \langle \boldsymbol{\psi}, \nabla_M \cdot \mathcal{F}((\boldsymbol{\phi}, \mathbf{m})) \rangle \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \quad (3.5)$$

La première composante de la dérivée du Lagrangien, obtenue de la dérivée par rapport à la variable adjointe, donne les équations aux dérivées partielles du modèle direct. La deuxième composante, obtenue de la dérivée du Lagrangien par rapport à la variable d'état, donne les équations aux dérivées partielles de la variable adjointe. Lorsque la variable d'état et la variable adjointe ont été calculées, le gradient de la fonction coût par rapport aux paramètres d'inversion peut être obtenu avec les équations données à la troisième composante de la dérivée du Lagrangien. Cette dernière composante est obtenue par la dérivée du Lagrangien par rapport aux paramètres d'inversion. Notons que l'opérateur $\mathcal{F}^*(\boldsymbol{\psi}, \mathbf{m})$ est l'adjoint de l'opérateur $\mathcal{F}(\boldsymbol{\phi}, \mathbf{m})$. L'adjoint d'un opérateur est

défini à partir du produit scalaire selon

$$\langle \boldsymbol{\psi}, \mathcal{F}(\boldsymbol{\phi}) \rangle = \langle \mathcal{F}^*(\boldsymbol{\psi}), \boldsymbol{\phi} \rangle. \quad (3.6)$$

Bref, la procédure suivante peut être utilisée pour dériver le modèle adjoint d'un système d'équations et obtenir le gradient de la fonction coût par rapport aux paramètres d'inversion:

1. Définir un modèle direct $\mathcal{F}(\boldsymbol{\phi}, \mathbf{m}) = 0$,
2. Définir une fonction coût $\mathcal{X}(\boldsymbol{\phi}, \mathbf{m})$,
3. Déterminer l'adjoint de l'opérateur direct $\mathcal{F}^*(\boldsymbol{\psi})$ à partir de la définition de l'équation (3.6),
4. Construire l'équation adjointe $\mathcal{F}^*(\boldsymbol{\psi}, \mathbf{m}) = \nabla_{\boldsymbol{\phi}} \mathcal{X}(\boldsymbol{\phi}, \mathbf{m})$,
5. Obtenir l'expression du gradient de la fonction coût: $\nabla \mathcal{X}|_{m_i} = \frac{\partial \mathcal{X}(\boldsymbol{\phi}, \mathbf{m})}{\partial m_i} - \left\langle \boldsymbol{\psi}, \frac{\partial \mathcal{F}(\boldsymbol{\phi}, \mathbf{m})}{\partial m_i} \right\rangle$.

En résumé, la méthode adjointe permet d'obtenir le gradient de la fonction coût en résolvant deux systèmes d'équations aux dérivées partielles: un système pour la variable d'état et un système pour la variable adjointe. Une fois celles-ci connues, le gradient de la fonction coût peut être calculé par leur produit scalaire. Comme les deux systèmes d'équations aux dérivées partielles sont habituellement semblables, le coût du calcul du gradient est habituellement de l'ordre de deux fois le coût de la modélisation directe.

3.2 Opérateurs auto-adjoints

Les opérateurs dits auto-adjoints sont particulièrement intéressants en inversion. Un opérateur est auto-adjoint lorsque l'opérateur et son adjoint sont équivalents: $\mathcal{F}(\cdot) = \mathcal{F}^*(\cdot)$. La définition d'un opérateur adjoint donnée à l'équation (3.6) permet de mieux comprendre ce que ceci implique. En particulier, dans ce travail, le produit scalaire suivant est adopté:

$$\langle \boldsymbol{\psi}(\mathbf{x}, t), \boldsymbol{\phi}(\mathbf{x}, t) \rangle = \int_T \int_X \psi_{\alpha}(\mathbf{x}, t) \phi_{\alpha}(\mathbf{x}, t) d\mathbf{x} dt. \quad (3.7)$$

Ce produit scalaire contient deux parties: une sommation sur les indices discrets α , d'un nombre égal aux nombres de variables sismiques (9+6L en 3D), et une intégration dans le temps et l'espace. En combinant la définition d'un opérateur auto-adjoint avec ce produit scalaire, et en nous limitant

à un opérateur linéaire \mathbf{A} , on obtient

$$\int_T \int_X \psi_\alpha(\mathbf{x}, t) A_{\alpha\beta} \phi_\beta(\mathbf{x}, t) d\mathbf{x} dt = \int_T \int_X A_{\alpha\beta} \psi_\alpha(\mathbf{x}, t) \phi_\beta(\mathbf{x}, t) d\mathbf{x} dt. \quad (3.8)$$

Ceci montre que sur les dimensions discrètes et réelles, un opérateur auto-adjoint est symétrique, c'est-à-dire $A_{\alpha\beta} = A_{\beta\alpha}$. De plus, dans ce travail, les éléments des opérateurs ne sont fonction du temps et de l'espace que par des premières et secondes dérivées. Dans ce cas, pour que l'opérateur soit auto-adjoint, la dérivation doit pouvoir commuter

$$\int_T \int_X \psi_\alpha(\mathbf{x}, t) \partial_i \phi_\beta(\mathbf{x}, t) d\mathbf{x} dt = \int_T \int_X \partial_i \psi_\alpha(\mathbf{x}, t) \phi_\beta(\mathbf{x}, t) d\mathbf{x} dt. \quad (3.9)$$

Ceci suggère une approche élégante pour dériver les équations adjointes impliquant des opérateurs linéaires quelconques:

1. Transformer les opérateurs en opérateurs auto-adjoints en appliquant des transformations linéaires,
2. Dériver les équations adjointes du système transformé,
3. Appliquer les transformations inverses aux équations adjointes

Cette dernière étape permet en particulier d'obtenir des équations directes et adjointes symétriques, c'est-à-dire qui partagent les mêmes équations aux dérivées partielles. En effet, si \mathbf{A} est un opérateur linéaire qui transforme l'opérateur directe \mathbf{F} en opérateur auto-adjoint, le Lagrangien de l'opérateur transformé mène aux équations adjointes suivantes:

$$\mathbf{A}\mathbf{F}\psi = \nabla_\Phi \mathcal{X}(\phi, \mathbf{m}). \quad (3.10)$$

L'application des transformations inverses mène à

$$\mathbf{F}\psi = \mathbf{A}^{-1} \nabla_\Phi \mathcal{X}(\phi, \mathbf{m}), \quad (3.11)$$

qui possède la même forme que le système directe non transformé ($\mathbf{F}\phi = \mathbf{s}$). Le terme source des équations adjointes est affecté par la transformation inverse, ce qui a peu de conséquences en pratique. De plus, l'expression du gradient de la fonction coût devient dans ce cas:

$$\nabla \mathcal{X}|_{m_i} = \frac{\partial \mathcal{X}(\phi, \mathbf{m})}{\partial m_i} - \left\langle \psi, \frac{\partial (\mathbf{A}\mathbf{F})\phi}{\partial m_i} \right\rangle. \quad (3.12)$$

3.3 Application au régime anisotrope viscoélastique

L'approche décrite précédemment est ici appliquée pour obtenir les équations adjointes de l'équation d'onde viscoélastique anisotrope. L'équation d'onde anisotrope viscoélastique donnée par l'équation (2.24) est

$$\begin{aligned}
\partial_t v_i - \frac{1}{\rho} D_{Ii} \sigma_I - s_{v_i} &= 0, \\
\partial_t \sigma_I - \Psi_{IJ}^0 D_{Jk} v_k - \sum_{l=1}^{L} r_I^l - s_{\sigma_I} &= 0, \\
\partial_t r_I^l + \frac{1}{\tau_{\sigma l}} \Psi_{IJ}^l D_{Jk} v_k + \frac{1}{\tau_{\sigma l}} r_I^l &= 0.
\end{aligned} \tag{3.13}$$

En notation matricielle, ce système d'équations peut s'écrire comme

$$\begin{bmatrix} \mathbf{I}_3 \partial_t & -\frac{1}{\rho} \mathbf{D}^T & 0 & \cdots & 0 \\ -\Psi_0 \mathbf{D} & \mathbf{I}_6 \partial_t & -\mathbf{I}_6 & \cdots & -\mathbf{I}_6 \\ \Psi_1 \mathbf{D} & 0 & \mathbf{I}_6 (\partial_t + \frac{1}{\tau_1}) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \Psi_L \mathbf{D} & 0 & 0 & \cdots & \mathbf{I}_6 (\partial_t + \frac{1}{\tau_L}) \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\sigma} \\ \mathbf{r}_1 \\ \vdots \\ \mathbf{r}_L \end{bmatrix} = \begin{bmatrix} \mathbf{s}_v \\ \mathbf{s}_\sigma \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}, \tag{3.14}$$

ou sous forme compacte

$$\mathbf{F} \boldsymbol{\phi} = \mathbf{s}. \tag{3.15}$$

À remarquer que l'opérateur \mathbf{F} n'est pas symétrique, et, de ce fait, n'est pas auto-adjoint. De plus, il n'est pas évident comment ce dernier pourrait devenir symétrique à l'aide de simples transformations linéaires.

Variabes de mémoire alternatives

En observant de plus près le système d'équations (3.13), il est possible de remarquer que la dépendance aux vitesses est tout à fait similaire pour les équations des contraintes et des variables de mémoire. Cependant, les équations des vitesses ne dépendent que des contraintes, ce qui brise la symétrie du système. Pour y pallier, il faudrait pouvoir transférer les termes des variables de mémoire des équations des contraintes aux équations des vitesses. Ceci est possible en définissant

une variable de mémoire alternative:

$$\partial_t \mathbf{R}^l = \mathbf{r}^l. \quad (3.16)$$

En utilisant la définition des variables de mémoire alternatives, le système d'équations (3.13) devient

$$\begin{aligned} \partial_t v_i - \frac{1}{\rho} D_{Ii} \sigma_I - s_{v_i} &= 0, \\ \partial_t \sigma_I - \Psi_{IJ}^0 D_{Jk} v_k - \sum_{l=1}^{l=L} \partial_t R_I^l - s_{\sigma_I} &= 0, \\ \partial_{tt} R_I^l + \frac{1}{\tau_{\sigma l}} \Psi_{IJ}^l D_{Jk} v_k + \frac{1}{\tau_{\sigma l}} \partial_t R_I^l &= 0. \end{aligned} \quad (3.17)$$

Finalement, nous définissons une contrainte modifiée, $\sigma'_I = \sigma_I - \sum_{l=1}^{l=L} R_I^l$, ce qui permet de récrire l'expression précédente comme

$$\begin{aligned} \partial_t v_i - \frac{1}{\rho} D_{Ii} \sigma'_I - \frac{1}{\rho} D_{Ii} \sum_{l=1}^{l=L} R_I^l - s_{v_i} &= 0, \\ \partial_t \sigma'_I - \Psi_{IJ}^0 D_{Jk} v_k - s_{\sigma_I} &= 0, \\ -\partial_t R_I^l - \Psi_{IJ}^l D_{Jk} v_k - \tau_{\sigma l} \partial_{tt} R_I^l &= 0. \end{aligned} \quad (3.18)$$

Sous cette forme, l'équation des vitesses présente la même dépendance aux contraintes modifiées qu'aux variables de mémoire, tandis que les équations des contraintes et des variables de mémoires contiennent la même forme de dépendance en vitesse. Notons que la formulation aux contraintes et variables de mémoire modifiées de l'équation (3.18) est tout à fait équivalente à l'équation standard (3.13). Cette équivalence est illustrée à la Figure 3.2, dans laquelle la solution par différences finies des deux équations est comparée pour un modèle constant de $V_p = 3500$ m/s, $V_s = 2000$ m/s, $\rho = 2000$ kg/m³ et $\tau_p = \tau_s = 0.2$. Les solutions pour les vitesses sont identiques à l'intérieur de la précision numérique des calculs (simple précision), de même que pour les contraintes.

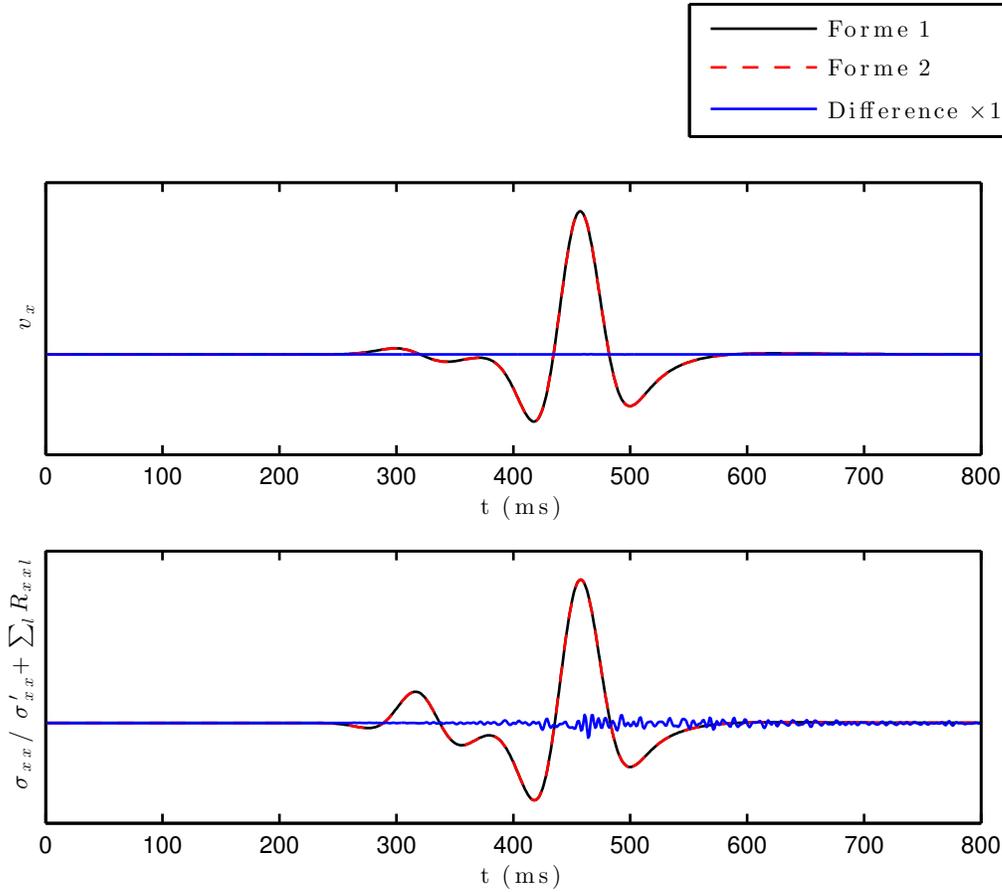


Figure 3.2 – Comparaison numérique entre la forme traditionnelle (forme 1) de l'équation d'onde et la forme utilisant la variable de mémoire alternative (forme 2).

Transformation auto-adjointe

Il est maintenant possible d'obtenir des opérateurs auto-adjoints à l'aide de transformations linéaires. Premièrement, le vecteur d'état et d'état adjoint sont définis comme:

$$\begin{aligned}\phi &= \left(v^T, \sigma^T, \mathbf{R}^1{}^T, \dots, \mathbf{R}^L{}^T \right)^T, \\ \psi &= \left(\tilde{v}^T, \tilde{\sigma}^T, \tilde{\mathbf{R}}^1{}^T, \dots, \tilde{\mathbf{R}}^L{}^T \right)^T,\end{aligned}\tag{3.19}$$

où $\tilde{\cdot}$ indique une variable adjointe. Le système d'équations (3.18) peut ainsi s'écrire en notation matricielle comme

$$\mathbf{A}\phi + \mathbf{B}\psi - \mathbf{G}\mathbf{C}\phi - \mathbf{s} = 0,\tag{3.20}$$

avec

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_9 & \\ & -\mathbf{I}_{6L} \end{bmatrix} \tau_{\sigma l} \partial_{tt}, \quad (3.21a)$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{I}_9 & \\ & -\mathbf{I}_{6L} \end{bmatrix} \partial_t, \quad (3.21b)$$

$$\mathbf{GC} = \begin{bmatrix} \frac{1}{\rho} \mathbf{I}_3 & \dots & 0 \\ & \Psi^0 & \vdots \\ \vdots & & \ddots \\ 0 & \dots & \Psi^L \end{bmatrix} \begin{bmatrix} \mathbf{0}_3 & D^T & \dots & D^T \\ D & & & \\ \vdots & & \mathbf{0}_{6+6L} & \\ D & & & \end{bmatrix}. \quad (3.21c)$$

Les opérateurs \mathbf{A} , \mathbf{B} et \mathbf{C} sont tous symétriques. Cependant, le produit \mathbf{GC} ne l'est pas. L'opérateur \mathbf{G} est bloc diagonal, chaque bloc contenant soit l'inverse de la densité, soit les tenseurs de rigidité. En se rappelant que le tenseur de rigidité est inversible, son inverse étant le tenseur des souplesses, l'inverse de l'opérateur \mathbf{G} doit exister aussi. Ainsi, il est possible de réécrire l'équation (3.20) comme

$$\mathbf{G}^{-1} \mathbf{A} \phi + \mathbf{G}^{-1} \mathbf{B} \phi - \mathbf{C} \phi - \mathbf{G}^{-1} \mathbf{s} = 0. \quad (3.22)$$

Les opérateurs transformés sont ainsi tous symétriques:

$$\mathbf{G}^{-1} \mathbf{A} = \begin{bmatrix} \mathbf{0}_3 & & \dots & 0 \\ & \mathbf{0}_6 & & \vdots \\ & & -\Psi^{1-1} & \\ \vdots & & & \ddots \\ 0 & \dots & & -\Psi^{L-1} \end{bmatrix} \tau_{\sigma l} \partial_{tt}, \quad (3.23a)$$

$$\mathbf{G}^{-1} \mathbf{B} = \begin{bmatrix} \rho \mathbf{I}_3 & & \dots & 0 \\ & \Psi^{0-1} & & \vdots \\ & & -\Psi^{1-1} & \\ \vdots & & & \ddots \\ 0 & \dots & & -\Psi^{L-1} \end{bmatrix} \partial_t, \quad (3.23b)$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{D}^T & \dots & \mathbf{D}^T \\ \mathbf{D} & & & \\ \vdots & & \mathbf{0}_{6+6L} & \\ \mathbf{D} & & & \end{bmatrix}. \quad (3.23c)$$

Le premier critère des opérateurs auto-adjoints est donc respecté. Le second critère concerne la commutation des dérivées partielles dans le temps et l'espace. La dérivation par partie est utilisée pour montrer cette propriété, ce qui donne

$$\int_{\Omega} \partial_i \psi_{\alpha} \phi_{\beta} d\Omega = \int_{\Gamma} \psi_{\alpha} \phi_{\beta} \hat{n}_i d\Gamma - \int_{\Omega} \psi_{\alpha} \partial_i \phi_{\beta} d\Omega, \quad (3.24)$$

où Ω est l'espace \mathbb{R}^4 contenant les dimensions temporelles et spatiales, Γ est la surface contenant les limites spatiales et temporelles \hat{n}_i est la $i^{\text{ème}}$ composante du vecteur unitaire normal à Γ . Ainsi, pour que les dérivées partielles commutent, l'intégration sur la surface doit être nulle. Ceci peut être obtenu en utilisant des conditions limites appropriées, par exemple:

- $\phi|_{t=0} = \partial_t \phi|_{t=0} = 0$ (passé nul),
- $\psi|_{t=T} = \partial_t \psi|_{t=T} = 0$ (futur nul),
- $\phi|_{\mathbf{x} \rightarrow \infty} = \psi|_{\mathbf{x} \rightarrow \infty} = 0$ (limites absorbantes).

Avec ces conditions limites, les dérivées commutent lors du produit scalaire. Ainsi, les opérateurs transformés ont les propriétés suivantes:

$$\begin{aligned} \langle \mathbf{G}^{-1} \mathbf{A} \psi, \phi \rangle &= \langle \psi, \mathbf{G}^{-1} \mathbf{A} \phi \rangle, \\ - \langle \mathbf{G}^{-1} \mathbf{B} \psi, \phi \rangle &= \langle \psi, \mathbf{G}^{-1} \mathbf{B} \phi \rangle, \\ - \langle \mathbf{C} \psi, \phi \rangle &= \langle \psi, \mathbf{C} \phi \rangle, \end{aligned} \quad (3.25)$$

c'est-à-dire que $\mathbf{G}^{-1} \mathbf{A}$ est auto-adjoint, et $\mathbf{G}^{-1} \mathbf{B}$ et \mathbf{C} sont anti-auto-adjoint. Notons que l'opérateur viscoélastique complet $\mathbf{G}^{-1} \mathbf{A} + \mathbf{G}^{-1} \mathbf{B} - \mathbf{C}$ n'est ni auto-adjoint, ni anti-auto-adjoint.

Dérivation des équations adjointes

Suite à ces transformations, les équations adjointes peuvent être obtenues facilement. Premièrement, le Lagrangien est défini par

$$\mathcal{L}(\tilde{\phi}, \tilde{\psi}; \mathbf{m}) = \mathcal{X}(\tilde{\phi}; \mathbf{m}) - \langle \tilde{\psi}, \mathbf{G}^{-1} \mathbf{A} \tilde{\phi} + \mathbf{G}^{-1} \mathbf{B} \tilde{\phi} - \mathbf{C} \tilde{\phi} - \mathbf{G}^{-1} \mathbf{s} \rangle. \quad (3.26)$$

En utilisant les propriétés adjointes des opérateurs, le Lagrangien peut être exprimé comme

$$\mathcal{L}(\tilde{\phi}, \tilde{\psi}; \mathbf{m}) = \mathcal{X}(\tilde{\phi}; \mathbf{m}) - \langle \mathbf{G}^{-1} \mathbf{A} \tilde{\psi} - \mathbf{G}^{-1} \mathbf{B} \tilde{\psi} + \mathbf{C} \tilde{\psi}, \tilde{\phi} \rangle + \langle \tilde{\psi}, \mathbf{G}^{-1} \mathbf{s} \rangle. \quad (3.27)$$

En égalant à 0 la dérivée de l'équation (3.27) par rapport à $\tilde{\phi}$, les équations adjointes pour le modèle direct transformé sont obtenues:

$$\frac{\partial \mathcal{X}}{\partial \phi} - \mathbf{G}^{-1} \mathbf{A} \psi + \mathbf{G}^{-1} \mathbf{B} \psi - \mathbf{C} \psi = 0. \quad (3.28)$$

En multipliant par \mathbf{G} cette expression, une forme plus près du modèle direct est obtenue, soit

$$\tilde{\mathcal{F}}(\psi, \phi; \mathbf{m}) = \mathbf{A} \psi - \mathbf{B} \psi + \mathbf{G} \mathbf{C} \psi - \mathbf{G} \frac{\partial \mathcal{X}}{\partial \phi} = 0. \quad (3.29)$$

Pour mieux voir la ressemblance entre les équations directes et adjointes, nous revenons à la notation indicielle:

$$\begin{aligned} -\partial_t \tilde{v}_i + \frac{1}{\rho} D_{Ii} \tilde{\sigma}'_I + \frac{1}{\rho} D_{Ii} \sum_{l=1}^{l=L} \tilde{R}_I^l - \frac{1}{\rho} \frac{\partial \mathcal{X}}{\partial v_i} &= 0, \\ -\partial_t \tilde{\sigma}'_I + \Psi_{IJ}^0 D_{Jk} \tilde{v}_k - \Psi_{IJ}^0 \frac{\partial \mathcal{X}}{\partial \sigma_J} &= 0, \\ \partial_t \tilde{R}_I^l + \Psi_{IJ}^l D_{Jk} \tilde{v}_k - \tau_{\sigma l} \partial_{tt} \tilde{R}_I^l &= 0. \end{aligned} \quad (3.30)$$

Comme les conditions limites des variables adjointes sont définies pour le temps final T , la variable temporelle doit être renversée ($t' = T - t$):

$$\begin{aligned} \partial_{t'} \tilde{v}_i + \frac{1}{\rho} D_{Ii} \tilde{\sigma}'_I + \frac{1}{\rho} D_{Ii} \sum_{l=1}^{l=L} \tilde{R}'_I{}^l - \frac{1}{\rho} \frac{\partial \mathcal{X}}{\partial v_i} &= 0, \\ \partial_{t'} \tilde{\sigma}'_I + \Psi_{IJ}^0 D_{Jk} \tilde{v}_k - \Psi_{IJ}^0 \frac{\partial \mathcal{X}}{\partial \sigma_J} &= 0, \\ -\partial_{t'} \tilde{R}'_I{}^l + \Psi_{IJ}^l D_{Jk} \tilde{v}_k - \tau_{\sigma l} \partial_{t'} \tilde{R}'_I{}^l &= 0. \end{aligned} \quad (3.31)$$

En utilisant la définition des contraintes modifiées, $\tilde{\sigma} = \tilde{\sigma}' + \sum_{l=1}^{l=L} \tilde{R}'^l$, et $\partial_{t'} \tilde{\sigma}' = \partial_{t'} \tilde{\sigma} - \sum_{l=1}^{l=L} \partial_{t'} \tilde{R}'^l$:

$$\begin{aligned} \partial_{t'} \tilde{v}_i + \frac{1}{\rho} D_{Ii} \tilde{\sigma}_I - \frac{1}{\rho} \frac{\partial \mathcal{X}}{\partial v_i} &= 0, \\ \partial_{t'} \tilde{\sigma}_I + \Psi_{IJ}^0 D_{Jk} \tilde{v}_k - \sum_{l=1}^{l=L} \partial_{t'} \tilde{R}'_I{}^l - \Psi_{IJ}^0 \frac{\partial \mathcal{X}}{\partial \sigma_J} &= 0, \\ -\partial_{t'} \tilde{R}'_I{}^l + \Psi_{IJ}^l D_{Jk} \tilde{v}_k - \tau_{\sigma l} \partial_{t'} \tilde{R}'_I{}^l &= 0. \end{aligned} \quad (3.32)$$

Finalement, revenant aux variables de mémoire traditionnelles $\tilde{r}'_I{}^l = -\partial_{t'} \tilde{R}'_I{}^l$:

$$\begin{aligned} \partial_{t'} \tilde{v}_i + \frac{1}{\rho} D_{Ii} \tilde{\sigma}_I - \frac{1}{\rho} \frac{\partial \mathcal{X}}{\partial v_i} &= 0, \\ \partial_{t'} \tilde{\sigma}_I + \Psi_{IJ}^0 D_{Jk} \tilde{v}_k + \sum_{l=1}^{l=L} \tilde{r}'_I{}^l - \Psi_{IJ}^0 \frac{\partial \mathcal{X}}{\partial \sigma_J} &= 0, \\ \partial_{t'} \tilde{r}'_I{}^l + \frac{1}{\tau_{\sigma l}} \Psi_{IJ}^l D_{Jk} \tilde{v}_k + \frac{1}{\tau_{\sigma l}} \tilde{r}'_I{}^l &= 0. \end{aligned} \quad (3.33)$$

Cette dernière équation pour l'état adjoint est identique au modèle direct de l'équation (3.13), à l'exception de changements de signe des dérivées spatiales et des termes des variables de mémoire dans l'expression des contraintes. De même, le terme source diffère et correspond aux résidus pour les équations adjointes. Comme les équations directes et adjointes partagent pratiquement le même système d'équations aux dérivées partielles, leur solution peut être obtenue par le même algorithme. Par contre, les variables adjointes sont calculées à rebours, du temps final au temps initial.

Finalement, le gradient de la fonction coût est donné par la dérivée du Lagrangien de l'équation (3.26) par rapport aux paramètres d'inversions,

$$\frac{\partial \mathcal{X}}{\partial m_\gamma} = - \left\langle \boldsymbol{\psi}, \frac{\partial \mathbf{G}^{-1}}{\partial m_\gamma} (\mathbf{A}\boldsymbol{\phi} + \mathbf{B}\boldsymbol{\phi} - \mathbf{s}) \right\rangle, \quad (3.34)$$

$$= - \left\langle \boldsymbol{\psi}, \frac{\partial \mathbf{G}^{-1}}{\partial m_\gamma} \mathbf{C} \boldsymbol{\phi} \right\rangle, \quad (3.35)$$

où la deuxième égalité vient de l'équation d'onde. En utilisant la définition du produit scalaire de l'équation (3.7), ceci devient:

$$\frac{\partial \mathcal{X}}{\partial m_\gamma} = - \int_T \int_X \psi_\alpha \frac{\partial G^{-1}_{\alpha\beta}}{\partial m_\gamma} (A_{\beta\lambda} \phi_\lambda + B_{\beta\lambda} \phi_\lambda - s_\beta) d\mathbf{x} dt. \quad (3.36)$$

Pour des paramètres ponctuels dans l'espace (par exemple, $m_\gamma = \rho(\mathbf{x})\delta(\mathbf{x})$), ceci représente la corrélation croisée dans le temps entre les variables d'état et les variables adjointes. En utilisant l'identité de Parseval, la corrélation dans le temps peut être transformée en une corrélation en fréquence:

$$\frac{\partial \mathcal{X}}{\partial m_\gamma} = - \frac{1}{2\pi} \int_\omega \int_X \hat{\psi}_\alpha \frac{\partial G^{-1}_{\alpha\beta}}{\partial m_\gamma} (\hat{A}_{\beta\lambda} \hat{\phi}_\lambda + \hat{B}_{\beta\lambda} \hat{\phi}_\lambda - \hat{s}_\beta) d\mathbf{x} d\omega, \quad (3.37)$$

où les variables avec un accent circonflexe sont les variables dans l'espace de Fourier. Il est alors possible d'évaluer la corrélation croisée fréquence par fréquence, ou avec une sélection de fréquences (voir chapitre suivant).

À noter aussi que le vecteur d'état inclut la variable de mémoire alternative \mathbf{R}^l et les contraintes modifiées $\boldsymbol{\sigma}'$. Si le code numérique est écrit avec les variables traditionnelles, il est possible de calculer les variables alternatives facilement en se basant sur leur définition.

3.4 Équivalence de la formulation vitesse-contrainte et de la formulation en déplacement

Tarantola (1988) et Tromp *et al.* (2004) obtiennent une équation d'onde viscoélastique anisotrope dans le formalisme du déplacement qui est complètement auto-adjointe, contrairement aux résultats présentés précédemment. Il est cependant possible de montrer qu'en transformant nos résultats dans la formulation du déplacement, une équation auto-adjointe est obtenue.

La composante des vitesses de l'équation adjointe (3.33) est dérivée dans le temps et la dérivée des contraintes est remplacée par son expression:

$$\begin{aligned} \partial_{t't'}\tilde{v}_i - \frac{1}{\rho}D_{Ji}(\Psi_{JK}^0 D_{Kk}\tilde{v}_k + \sum_{l=1}^{l=L}\tilde{r}_J^l) &= 0, \\ \partial_{t'}\tilde{r}_I^l + \frac{1}{\tau_{\sigma l}}\Psi_{IJ}^l D_{Jk}\tilde{v}_k + \frac{1}{\tau_{\sigma l}}\tilde{r}_I^l &= 0. \end{aligned} \quad (3.38)$$

En substituant les vitesses pour le déplacement $\tilde{v} = -\partial_{t'}\tilde{u}$ et les variables de mémoires $\tilde{r} = -\partial_{t'}\tilde{R}$:

$$\begin{aligned} -\partial_{t't't'}\tilde{u}_i - \frac{1}{\rho}D_{Ji}(-\Psi_{JK}^0 D_{Kk}\partial_{t'}\tilde{u}_k - \sum_{l=1}^{l=L}\partial_{t'}\tilde{R}_J^l) &= 0, \\ -\partial_{t't'}\tilde{R}_I^l - \frac{1}{\tau_{\sigma l}}\Psi_{IJ}^l D_{Jk}\partial_{t'}\tilde{u}_k - \frac{1}{\tau_{\sigma l}}\partial_{t'}\tilde{R}_I^l &= 0. \end{aligned} \quad (3.39)$$

et en intégrant ces deux équations dans le temps:

$$\begin{aligned} \partial_{t't'}\tilde{u}_i - \frac{1}{\rho}D_{Ji}\Psi_{JK}^0 D_{Kk}\tilde{u}_k - \frac{1}{\rho}D_{Ji}\sum_{l=1}^{l=L}\tilde{R}_J^l &= 0, \\ \partial_{t'}\tilde{R}_I^l + \frac{1}{\tau_{\sigma l}}\Psi_{IJ}^l D_{Jk}\tilde{u}_k + \frac{1}{\tau_{\sigma l}}\tilde{R}_I^l &= 0. \end{aligned} \quad (3.40)$$

La même démarche peut être adoptée pour l'équation directe (3.13), menant à l'expression suivante:

$$\begin{aligned} \partial_{tt}u_i - \frac{1}{\rho}D_{Ji}\Psi_{JK}^0 D_{Kk}u_k - \frac{1}{\rho}D_{Ji}\sum_{l=1}^{l=L}R_J^l &= 0, \\ \partial_t R_I^l + \frac{1}{\tau_{\sigma l}}\Psi_{IJ}^l D_{Jk}u_k + \frac{1}{\tau_{\sigma l}}R_I^l &= 0. \end{aligned} \quad (3.41)$$

Ces deux dernières équations sont identiques, et montrent que la démarche adoptée mène bien à une équation d'onde auto-adjointe dans la formulation du déplacement. Ceci montre aussi l'équivalence entre notre formulation, celle de Tarantola (1988), de Tromp *et al.* (2004) et de Yang *et al.* (2016b).

3.5 Application au régime isotrope viscoélastique

Comme la suite de travaux ne concerne que le cas viscoélastique isotrope, cette section s'attarde plus en détail à ce cas particulier. Dans le cas isotrope 3D, les matrices Ψ^l sont données par

$$\Psi_{3D}^l = \begin{bmatrix} \lambda^l + 2\mu^l & \lambda^l & \lambda^l & 0 & 0 & 0 \\ \lambda^l & \lambda^l + 2\mu^l & \lambda^l & 0 & 0 & 0 \\ \lambda^l & \lambda^l & \lambda^l + 2\mu^l & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu^l & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu^l & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu^l \end{bmatrix}, \quad (3.42)$$

et en 2D par:

$$\Psi_{2D}^l = \begin{bmatrix} \lambda^l + 2\mu^l & \lambda^l & 0 \\ \lambda^l & \lambda^l + 2\mu^l & 0 \\ 0 & 0 & \mu^l \end{bmatrix}. \quad (3.43)$$

La décomposition en valeurs propres de cette matrice donne

$$\Psi_{N_d}^l = \mathbf{R}_{N_d} \mathbf{\Lambda}_{N_d}^l \mathbf{R}_{N_d}. \quad (3.44)$$

En trois dimensions, les matrices propres et vecteurs propres sont donnés par

$$\mathbf{R}_{3D} = \begin{bmatrix} \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & 0 & 0 & 0 \\ \frac{1}{\sqrt{3}} & -\frac{1}{2}(\sqrt{3}+1) & \frac{1}{2}(\sqrt{3}-1) & 0 & 0 & 0 \\ \frac{1}{\sqrt{3}} & \frac{1}{2}(\sqrt{3}-1) & -\frac{1}{2}(\sqrt{3}+1) & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.45)$$

$$\mathbf{\Lambda}_{3D}^l = \begin{bmatrix} 3\lambda^l + 2\mu^l & 0 & 0 & 0 & 0 & 0 \\ 0 & 2\mu^l & 0 & 0 & 0 & 0 \\ 0 & 0 & 2\mu^l & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu^l & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu^l & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu^l \end{bmatrix}. \quad (3.46)$$

En 2D, celles-ci sont données par

$$\mathbf{R}_{2D} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (3.47)$$

$$\mathbf{\Lambda}_{2D}^l = \begin{bmatrix} 3\lambda^l + 2\mu^l & 0 \\ 0 & 2\mu^l & 0 \\ 0 & 0 & \mu^l \end{bmatrix}. \quad (3.48)$$

Ainsi, l'opérateur \mathbf{G} s'écrit comme

$$\begin{aligned} \mathbf{G} &= \mathbf{T}\mathbf{\Lambda}\mathbf{T} \\ &= \begin{bmatrix} \mathbf{I}_{N_d} & \cdots & 0 \\ & \mathbf{R}_{N_d} & \vdots \\ \vdots & & \ddots \\ 0 & \cdots & \mathbf{R}_{N_d} \end{bmatrix} \begin{bmatrix} \frac{1}{\rho}\mathbf{I}_{N_d} & \cdots & 0 \\ & \mathbf{\Lambda}_{N_d}^0 & \vdots \\ \vdots & & \ddots \\ 0 & \cdots & \mathbf{\Lambda}_{N_d}^L \end{bmatrix} \begin{bmatrix} \mathbf{I}_{N_d} & \cdots & 0 \\ & \mathbf{R}_{N_d} & \vdots \\ \vdots & & \ddots \\ 0 & \cdots & \mathbf{R}_{N_d} \end{bmatrix}. \end{aligned} \quad (3.49)$$

Son inverse est directement obtenu en prenant l'inverse des valeurs propres, soit

$$\mathbf{G}^{-1} = \mathbf{T}\mathbf{\Lambda}^{-1}\mathbf{T}. \quad (3.50)$$

L'équation (3.36) donnant le gradient de la fonction coût devient ainsi dans le cas isotrope

$$\frac{\partial \mathcal{X}}{\partial m_\alpha} = - \left\langle \boldsymbol{\psi}, \mathbf{T} \frac{\partial \mathbf{\Lambda}^{-1}}{\partial m_\alpha} \mathbf{T} (\mathbf{A}\boldsymbol{\phi} + \mathbf{B}\boldsymbol{\phi} - \mathbf{s}) \right\rangle. \quad (3.51)$$

En adoptant une paramétrisation en termes de modules des vitesses et de niveaux d'atténuation ($\mathbf{m} = [\rho, M, \mu, \tau_p, \tau_s]$), le gradient peut être explicité comme suit:

$$\frac{\partial \mathcal{X}}{\partial \rho} = \langle \bar{v}_x, \partial_t v_x \rangle + \langle \bar{v}_y, \partial_t v_y \rangle + \langle \bar{v}_z, \partial_t v_z \rangle, \quad (3.52a)$$

$$\frac{\partial \mathcal{X}}{\partial M} = -c_1^M P_1 + c_2^M P_2, \quad (3.52b)$$

$$\frac{\partial \mathcal{X}}{\partial \tau_p} = -c_1^{\tau_p} P_1 + c_2^{\tau_p} P_2, \quad (3.52c)$$

$$\frac{\partial \mathcal{X}}{\partial \mu} = -c_1^\mu P_3 + c_2^\mu P_1 - c_3^\mu P_4 + c_4^\mu P_5 - c_5^\mu P_2 + c_6^\mu P_6, \quad (3.52d)$$

$$\frac{\partial \mathcal{X}}{\partial \tau_s} = -c_1^{\tau_s} P_3 + c_2^{\tau_s} P_1 - c_3^{\tau_s} P_4 + c_4^{\tau_s} P_5 - c_5^{\tau_s} P_2 + c_6^{\tau_s} P_6, \quad (3.52e)$$

avec les produits scalaires suivants:

$$P_1 = \langle \bar{\sigma}'_{xx} + \bar{\sigma}'_{yy} + \bar{\sigma}'_{zz}, \partial_t(\sigma'_{xx} + \sigma'_{yy} + \sigma'_{zz}) \rangle, \quad (3.53a)$$

$$P_2 = \langle \bar{R}_{xxl} + \bar{R}_{yyt} + \bar{R}_{zzl}, (1 + \tau_{\sigma l} \partial_t)(r_{xxl} + r_{yyt} + r_{zzl}) \rangle, \quad (3.53b)$$

$$P_3 = \langle \bar{\sigma}'_{xy}, \partial_t \sigma'_{xy} \rangle + \langle \bar{\sigma}'_{xz}, \partial_t \sigma'_{xz} \rangle + \langle \bar{\sigma}'_{yz}, \partial_t \sigma'_{yz} \rangle, \quad (3.53c)$$

$$\begin{aligned} P_4 = & \langle \bar{\sigma}'_{xx}, \partial_t((N_d - 1)\sigma'_{xx} - \sigma'_{yy} - \sigma'_{zz}) \rangle \\ & + \langle \bar{\sigma}'_{yy}, \partial_t((N_d - 1)\sigma'_{yy} - \sigma'_{xx} - \sigma'_{zz}) \rangle \\ & + \langle \bar{\sigma}'_{zz}, \partial_t((N_d - 1)\sigma'_{zz} - \sigma'_{xx} - \sigma'_{yy}) \rangle, \end{aligned} \quad (3.53d)$$

$$P_5 = \langle \bar{R}_{xyt}, (1 + \tau_{\sigma l} \partial_t)r_{xyt} \rangle + \langle \bar{R}_{xzl}, (1 + \tau_{\sigma l} \partial_t)r_{xzl} \rangle + \langle \bar{R}_{yzt}, (1 + \tau_{\sigma l} \partial_t)r_{yzt} \rangle, \quad (3.53e)$$

$$\begin{aligned} P_6 = & \langle \bar{R}_{xxl}, (1 + \tau_{\sigma l} \partial_t)((N_d - 1)r_{xxl} - r_{yyt} - r_{zzl}) \rangle \\ & + \langle \bar{R}_{yyt}, (1 + \tau_{\sigma l} \partial_t)((N_d - 1)r_{yyt} - r_{xxl} - r_{zzl}) \rangle \\ & + \langle \bar{R}_{zzl}, (1 + \tau_{\sigma l} \partial_t)((N_d - 1)r_{zzl} - r_{xxl} - r_{yyt}) \rangle. \end{aligned} \quad (3.53f)$$

Les coefficients sont donnés par:

$$\begin{aligned}
c_1^M &= \frac{(1 + \tau_p)}{(1 + \alpha\tau_p)} b_1, & c_1^{\tau_p} &= (1 - \alpha) \frac{M}{(1 + \alpha\tau_p)^2} b_1, \\
c_2^M &= \frac{\tau_p}{(1 + \alpha\tau_p)} b_2, & c_2^{\tau_p} &= \frac{M}{(1 + \alpha\tau_p)^2} b_2, \\
c_1^\mu &= \frac{1}{\mu^2} \frac{(1 + \alpha\tau_s)}{(1 + \tau_s)}, & c_1^{\tau_s} &= \frac{1}{\mu} \frac{(1 - \alpha)}{(1 + \tau_s)^2}, \\
c_2^\mu &= \frac{N_d + 1}{3} \frac{(1 + \tau_s)}{(1 + \alpha\tau_s)} b_1, & c_2^{\tau_s} &= \frac{N_d + 1}{3} (1 - \alpha) \frac{\mu}{(1 + \alpha\tau_s)^2} b_1, \\
c_3^\mu &= \frac{1}{2N_d} \frac{1}{\mu^2} \frac{(1 + \alpha\tau_s)}{(1 + \tau_s)}, & c_3^{\tau_s} &= \frac{1 - \alpha}{2N_d \mu (1 + L\tau_s)^2}, \\
c_4^\mu &= \frac{1}{\mu^2} \frac{(1 + \alpha\tau_s)}{\tau_s}, & c_4^{\tau_s} &= \frac{1}{\mu\tau_s^2}, \\
c_5^\mu &= \frac{N_d + 1}{3} \frac{\tau_s}{(1 + \alpha\tau_s)} b_2, & c_5^{\tau_s} &= \frac{N_d + 1}{3} \frac{\mu}{(1 + \alpha\tau_s)^2} b_2, \\
c_6^\mu &= \frac{1}{2N_d} \frac{1}{\mu^2} \frac{(1 + \alpha\tau_s)}{\tau_s}, & c_6^{\tau_s} &= \frac{1}{2N_d \mu \tau_s^2}.
\end{aligned} \tag{3.54}$$

avec b_1 et b_2 définis comme

$$b_1 = \left(N_d M \frac{(1 + \tau_p)}{(1 + \alpha\tau_p)} - 2(N_d - 1) \mu \frac{(1 + \tau_s)}{(1 + \alpha\tau_s)} \right)^{-2}, \tag{3.55a}$$

$$b_2 = \left(N_d M \frac{\tau_p}{(1 + \alpha\tau_p)} - 2(N_d - 1) \mu \frac{\tau_s}{(1 + \alpha\tau_s)} \right)^{-2}. \tag{3.55b}$$

3.6 Conclusion

Les équations adjointes et le gradient de la fonction coût développés dans ce chapitre permettent de calculer efficacement la direction de descente requise dans les algorithmes de recherche linéaire pour l'équation viscoélastique anisotrope, et plus particulièrement l'équation isotrope. Étant donné que les équations directes et adjointes sont, à un signe près, identiques (équations (3.13) et (3.33)), le même code de modélisation par différences finies peut être utilisé pour obtenir les champs directs et adjoints. Ceci est un avantage indéniable, un code bien optimisé de modélisation étant coûteux à développer. Ce dernier point est l'objet du chapitre suivant.

Chapitre 4

Résolution par différences finies sur des plateformes de calcul hétérogènes

La résolution numérique de l'équation d'onde représente le principal goulot d'étranglement de l'inversion de formes d'ondes complètes. Celle-ci est requise tant pour le champ direct que pour le champ adjoint lors du calcul du gradient de la fonction coût. Ainsi, un code de modélisation sismique plus rapide permettrait de doubler le temps de calcul d'une itération d'inversion. Ce chapitre traite de l'utilisation de plateformes de calcul hétérogènes avec OpenCL pour l'inversion de formes d'ondes complètes viscoélastique isotrope selon la méthode adjointe développée au chapitre précédent. En particulier, un code complet permettant d'obtenir la solution de l'équation d'onde ainsi que le gradient de la fonction coût par la méthode adjointe est présenté en détail. Dans un premier temps, les motivations du calcul parallèle dans le contexte des différences finies sont abordées. Puis, les plateformes de calculs parallèles sont discutées, et une brève introduction à OpenCL est présentée. Certains détails du code sont ensuite discutés et les gains de performance sont testés dans divers environnements de calcul parallèle. Ces résultats font l'objet de la publication suivante:

- Fabien-Ouellet, G., Gloaguen, E., & Giroux, B. (2016). Time-domain seismic modeling in viscoelastic media for full waveform inversion on heterogeneous computing platforms with OpenCL. *Computers & Geosciences*. doi:10.1016/j.cageo.2016.12.004

4.1 Parallélisme des différences finies

Dans l'idéal, l'accélération obtenue de la parallélisation d'un programme informatique devrait être linéaire: en doublant les ressources en processeurs, le temps d'exécution devrait diminuer par deux. Or, en pratique, ceci n'est pas réaliste. En effet, ce ne sont pas toutes les sections d'un algorithme qui peuvent être calculées de façon parallèle et certains calculs nécessitent la complétion de calculs préalables. De plus, les variables en mémoire doivent être acheminées au processeur. Amdahl (1967) propose la loi suivante pour l'accélération d'une tâche à exécution constante:

$$S = \frac{1}{1 - p + \frac{p}{s}}, \quad (4.1)$$

où S est l'accélération de la tâche totale, s est l'accélération de la section parallélisable et p est la fraction parallélisable de la tâche. Cette loi reste très idéalisée et ne comprend pas, notamment, le temps de latence causé par les communications ou certains calculs supplémentaires nécessaires aux calculs parallèles. Ainsi, cette loi donne une limite supérieure de l'accélération obtenue par le calcul parallèle.

Néanmoins, il est intéressant de constater que les parties d'un programme ne pouvant être parallélisées induisent une limite supérieure de l'accélération due parallélisation. Ceci est illustré à la Figure 4.1, montrant l'accélération d'un programme en fonction du nombre de processeurs pour différentes fractions parallélisables. On constate que pour p inférieur à 1, l'accélération atteint un plateau. Ce plateau est en fait donné par $s \rightarrow \infty$:

$$S_{max} = \frac{1}{1 - p}. \quad (4.2)$$

Qu'en est-il de la modélisation sismique par différences finies? En particulier, quelle est la fraction parallélisable? Pour le calculer, reprenons la formule de mise à jour des vitesses des particules de l'équation (2.52), soit

$$v_x^{p,q,r,t+1} = v_x^{p,q,r,t} + \frac{\Delta t}{\rho_x^{p,q,r}} \left[D_x^+ \sigma_{xx}^{p,q,r,t-1/2} + D_y^- \sigma_{xy}^{p,q,r,t-1/2} + D_z^- \sigma_{xz}^{p,q,r,t-1/2} \right] + s_{v_x}^{p,q,r,t}. \quad (4.3)$$

Cette équation indique que pour chaque point de la grille p, q, r , la mise à jour au temps t peut être faite simultanément. Il en est de même pour la mise à jour des contraintes (éq. (2.50)) et des variables de mémoire (éq. (2.51)). Il est alors clair que l'accélération maximale S_{max} est proportionnelle

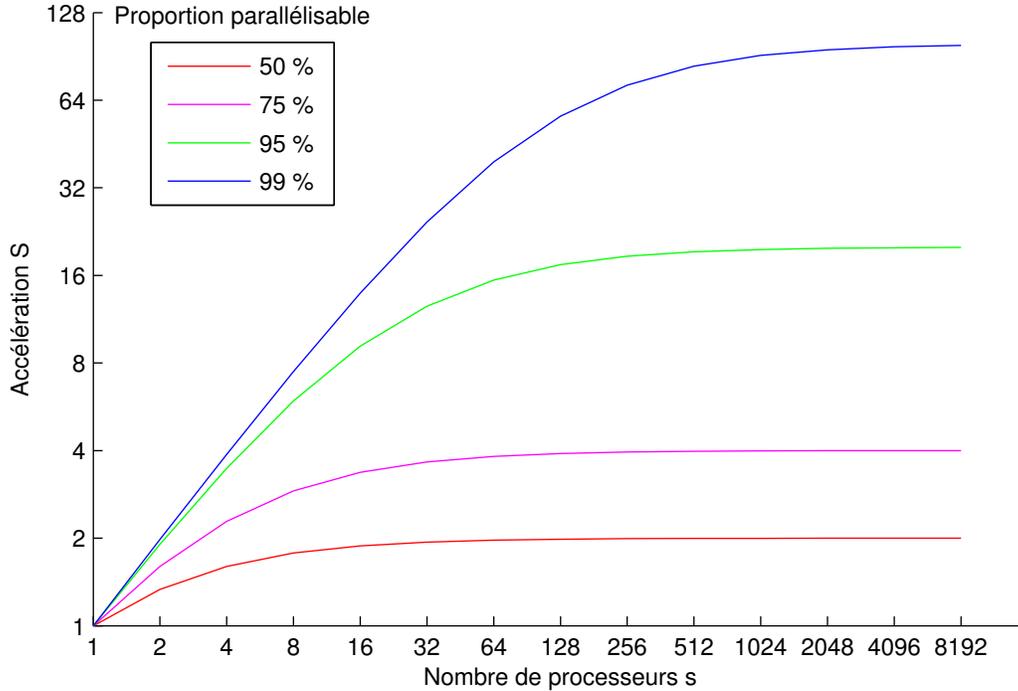


Figure 4.1 – Loi d’Amdhal pour différentes portions parallèles d’une tâche.

à $N_x N_y N_z$. En utilisant la formule précédente, il est possible d’obtenir une idée de la fraction parallélisable d’un programme en différences finies :

$$p = 1 - \frac{1}{N_x N_y N_z}. \quad (4.4)$$

Ainsi, plus la taille de la grille est grande, plus la fraction parallélisable augmente, et plus l’accélération maximale offerte par la parallélisation grandit. Notons que pour des modèles de grandes tailles comportant des milliards d’éléments, le gain maximal dépasse largement le nombre de processeurs accessibles sur les grappes de calcul modernes. Ceci est la principale force de la méthode des différences finies : pour des grilles de grandes tailles, l’accélération devrait en théorie varier linéairement avec le nombre de processeurs.

En pratique, le temps requis par les communications entre les processeurs introduit une limite à l’accélération possible. En effet, à chaque pas de temps, chaque processeur doit communiquer les valeurs qu’il a mises à jour aux autres processeurs, étant donné que le calcul des dérivées par différences finies requiert les valeurs aux points adjacents de la grille. Selon la rapidité des communications entre les processeurs où les engins de calcul, ceci peut affecter les performances de scalabilité.

À noter qu'une autre approche de parallélisme peut être adoptée pour la solution en différences finies de l'équation d'onde. En effet, au lieu de paralléliser la mise au jour des grilles sismiques, il est possible de paralléliser les simulations de chacun des tirs. En effet, la solution des équations (2.50), (2.51) et (2.52), est totalement indépendante pour chacun des points de tirs \mathbf{s} et aucune communication n'est requise entre les processeurs simulant différentes sources. Sachant que l'inversion de formes d'ondes complètes requiert habituellement plusieurs centaines de points de tir, la limite théorique donnée par la loi d'Amdahl est infinie, et donc une scalabilité parfaite peut être atteinte.

Le parallélisme des mises à jour demeure toutefois indispensable en raison des ressources de calcul gigantesques et de vastes quantités de mémoire requises pour l'inversion de formes d'ondes complètes viscoélastique en 3D. Aujourd'hui, ceci n'est possible qu'en partageant la mémoire et les calculs entre différents noeuds d'une grappe de calcul. Par exemple, une grille tridimensionnelle de 5000x5000x5000 éléments requiert 125×10^9 valeurs, c'est-à-dire 500 gigaoctets en mémoire pour une représentation en simple précision. Pour l'équation d'onde viscoélastique avec 3 corps de Maxwell, 27 variables dynamiques et 5 propriétés mécaniques sont requises. Par conséquent, 32 de ces grilles doivent être contenues en mémoire. Au total, 16 téraoctets de mémoires vives sont requis pour un tel modèle, ce qui excède les capacités des ordinateurs d'aujourd'hui.

4.2 Plateformes de calculs hétérogènes

Le recours aux grappes de calcul, c'est-à-dire une collection d'ordinateurs reliés par un réseau à haut débit, demeure ainsi essentiel. En effet, une grappe de calcul permet de partager les calculs ainsi que la mémoire requise entre plusieurs ordinateurs. Comme chaque ordinateur possède son propre espace mémoire et ne partage pas directement les instructions à effectuer, une approche de programmation spécifique au calcul parallèle sur réseau doit être adaptée. Pour ce faire, plusieurs stratégies existent, la plus courante en calcul scientifique étant sans contredit le *Message Passing Interface* ou MPI. Une librairie MPI permet en effet au programmeur d'écrire simplement les instructions pour la communication des données et la synchronisation des instructions entre les noeuds de calculs, ainsi qu'entre les différents coeurs d'un CPU (*Central Processing Units*).

L'approche du MPI pour un langage standard comme le C99 fonctionne bien pour une grappe de calcul homogène dans laquelle chacun des noeuds ne comprend que des CPUs d'une même architec-

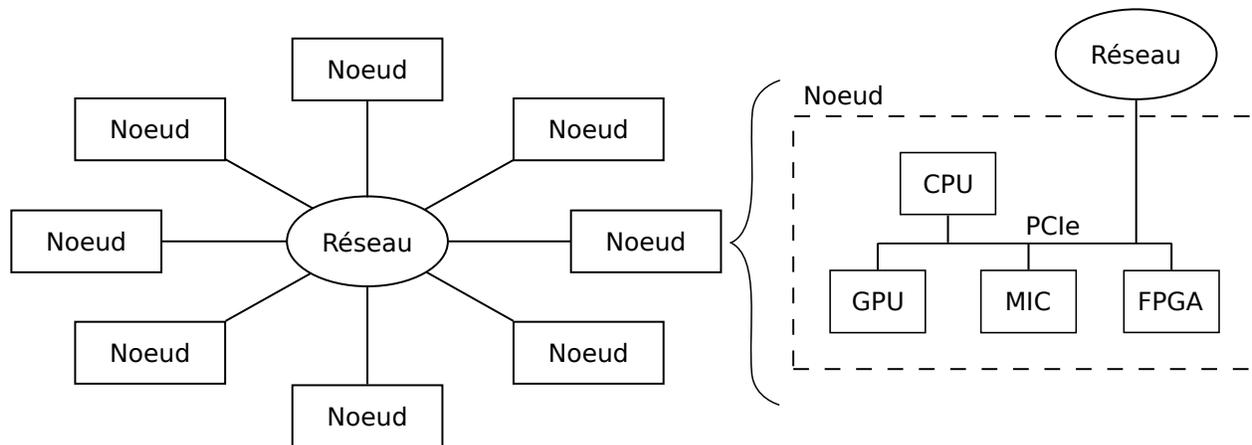


Figure 4.2 – Schématisation d’une grappe de calcul (gauche) et d’un noeud hétérogène de cette grappe (droite).

ture. Or, les grappes modernes sont de plus en plus hétérogènes, c’est-à-dire que chaque noeud peut contenir plusieurs types de processeurs en plus des CPU, notamment des cartes graphiques ou GPU (*Graphics Processing Units*), des FPGA (*Field-Programmable Gate Array*) ou des coprocesseurs. Ceci est schématisé à la Figure 4.2. Dans ce travail, nous adoptons un modèle dans lequel le CPU gère le système d’exploitation et peut accéder aux différents engins de calculs par un lien PCIe. Étant donné les débits de données limités pouvant passer à travers ces ports, un grand effort des manufacturiers comme Nvidia et Intel a été porté pour améliorer la rapidité des communications entre le CPU et les engins de calculs, et de créer un espace mémoire unifié entre les différents processeurs. L’évolution de ces technologies est très rapide et celles-ci ne sont pas encore très répandues, c’est pourquoi nous en ferons fi dans ce travail.

Une grappe de calcul hétérogène comporte plusieurs niveaux de parallélisme. Au premier niveau, il y a le parallélisme des noeuds: chaque noeud peut procéder à des calculs simultanément. Au deuxième niveau, il y a les engins de calculs que contient chacun des noeuds. Par exemple, chaque ordinateur peut contenir plusieurs CPUs qui procèdent simultanément à des calculs. Enfin, au troisième niveau, chaque processeur peut procéder parallèlement à plusieurs opérations à la fois, soit en utilisant plusieurs coeurs ou en vectorisant les opérations. Comme mentionné précédemment, l’interface MPI permet de gérer le parallélisme au niveau de la grappe, et à l’intérieur des noeuds lorsque des CPUs sont utilisés pour les calculs.

Ce troisième niveau de parallélisme est illustré à la Figure 4.3. À la base, un processeur à un coeur peut être compris selon l’architecture de von Neumann. Celle-ci contient trois éléments:

la mémoire, une unité arithmétique qui procède aux opérations mathématiques et une unité de contrôle qui gère les instructions à effectuer et la mémoire. Dans ce modèle, une opération par cycle peut être effectuée et les instructions sont donc opérées en série. Les processeurs modernes sont cependant beaucoup plus complexes que ce modèle rudimentaire. Ainsi, un CPU moderne contient plusieurs coeurs, de l'ordre d'une dizaine. Chaque coeur comprend une unité de contrôle capable de gérer indépendamment une série d'instructions. Chacun des coeurs peut comprendre plusieurs unités arithmétiques et logiques, et donc exécuter plusieurs opérations à la fois. De même, plusieurs niveaux de mémoire existent: la mémoire cache propre à chacun des coeurs et la mémoire vive partagées par tous les coeurs du CPU. L'architecture d'une carte graphique n'est pas si différente de celle d'un CPU. Ainsi, un GPU contient aussi plusieurs coeurs contenant chacun une unité de contrôle. Chaque coeur comprend aussi une mémoire locale et plusieurs unités arithmétiques. La principale différence entre ces deux architectures est une différence d'optimisation de conception: le ratio unité arithmétique par coeur du GPU est beaucoup plus élevé, ce qui lui permet d'effectuer plus d'opérations par cycle que le CPU, mais enlève de la souplesse quant aux instructions pouvant être envoyées à chaque unité arithmétique. Le GPU est ainsi plutôt pensé pour un modèle d'instruction unique, mémoire multiple où chacune des unités arithmétiques procède aux mêmes opérations sur différentes adresses de mémoire. Le CPU quant à lui permet d'effectuer plusieurs instructions différentes à la fois à une plus grande vitesse de cycle que le GPU.

4.3 Calcul scientifique sur cartes graphiques

Les GPUs incarnent donc la tendance du parallélisme des processeurs à son extrême; un seul GPU peut contenir plusieurs milliers de fils d'exécution. À la base, ceux-ci sont conçus pour l'affichage graphique, permettant en particulier aux jeux vidéos d'offrir un rendu 3D toujours plus réaliste. Les GPUs peuvent néanmoins être utilisés pour effectuer des calculs numériques scientifiques. Ainsi, lorsque l'algorithme s'y prête, l'utilisation de cartes graphiques peut accélérer les simulations numériques de plusieurs ordres de grandeur en comparaison au même code utilisant un CPU standard. Ceci a causé une explosion de l'utilisation des cartes graphiques pour le calcul scientifique (Nickolls & Dally, 2010), de la bio-informatique à l'astrophysique de grande échelle. De plus, les cartes graphiques sont de plus en plus utilisées dans des larges grappes de calcul (Zhe *et al.*, 2004), et la plupart des superordinateurs les plus rapides au monde intègrent des cartes

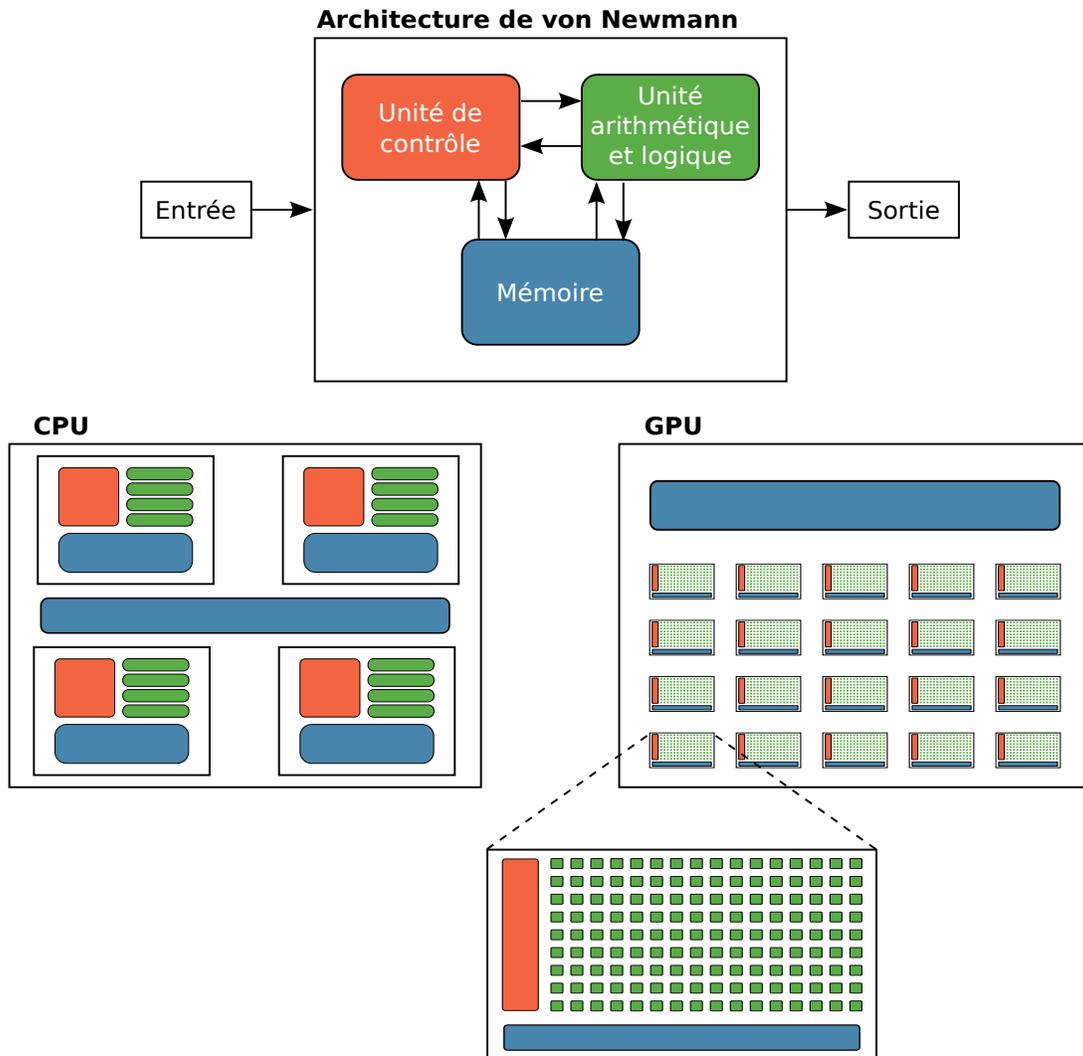


Figure 4.3 – Architecture de von Neumann (haut), architecture conceptuelle d’un CPU à 4 coeurs (bas, gauche) et d’un GPU avec 20 coeurs (bas, droite).

graphiques ou des accélérateurs (Meuer *et al.*, 2016). Cependant, les cartes graphiques ne sont pas avantageuses pour n’importe quel type de calcul scientifique (Vuduc *et al.*, 2010): seuls les calculs suivant le modèle de programmation du programme-unique données-multiples y sont efficaces.

En géophysique, et plus précisément en sismologie, le calcul GPU a été appliqué avec le plus de succès pour la résolution de l’équation d’onde par la méthode des différences finies dans le domaine du temps (FDTD). Comme il a été montré à la section précédente, la méthode des différences finies s’adapte bien au calcul parallèle sur GPU, étant donné que les mises à jour de chaque point de la grille spatiale sont indépendantes les unes des autres et s’opèrent selon les mêmes instructions. Par exemple, pour l’équation d’onde élastique, Michéa & Komatitsch (2010) obtiennent une diminution

des temps de calcul d'un facteur entre 20 et 60 avec un GPU par rapport aux calculs effectués sur un seul coeur d'un CPU. Okamoto *et al.* (2011) obtiennent une accélération 45 fois supérieure pour une implémentation avec un seul GPU et présentent une implémentation multi-GPU qui parallélise les calculs sur plusieurs GPU, avec une scalabilité sous-linéaire. Rubio *et al.* (2014) et Weiss & Shragge (2013) présentent tous deux des implémentations en différences finies multi-GPU pour l'équation d'onde élastique anisotrope qui montrent la même scalabilité sous-linéaire. Le partage des calculs entre plusieurs GPUs par la décomposition de domaine semble être problématique en raison de la lenteur des communications de mémoire entre le GPU et le CPU. Le calcul GPU a aussi été appliqué avec succès à la méthode des éléments spectraux (Komatitsch *et al.*, 2010), la méthode de Galerkin discontinue (Mu *et al.*, 2013) et la migration à temps inverse (Abdelkhalek *et al.*, 2009), entre autres.

La grande majorité des implémentations GPUs en modélisation sismique sont écrites dans l'interface de programmation CUDA (Nvidia, 2007). CUDA simplifie grandement les efforts de programmation scientifique pour les cartes graphiques Nvidia. Cependant, un code écrit en CUDA n'est compatible qu'avec les cartes graphiques Nvidia. Son utilisation est donc un acte de foi que les cartes Nvidia sont, et demeureront les processeurs les plus efficaces pour les calculs scientifiques. De plus, plusieurs grappes de calculs offrent différents types de processeurs (CPU, GPU, accélérateurs, etc). Ainsi, le choix de CUDA ne permet pas de profiter pleinement des capacités de calculs offertes par ces grappes hétérogènes. Par ailleurs, OpenCL (Stone *et al.*, 2010) est une interface de programmation *open source* compatible avec la plupart des types de processeurs sur le marché: cartes graphiques, CPU, accélérateurs et circuits logiques programmables (FPGA), entre autres. De plus, les performances de OpenCL et de CUDA sont comparables sur les cartes graphiques Nvidia (Du *et al.*, 2012). Malgré cet avantage, peu de codes de modélisation sismique adoptent OpenCL: Iturrarán-Viveros & Molero (2013) utilisent OpenCL pour modéliser des simulations acoustiques 2.5D, Kloc & Danek (2013) utilisent OpenCL pour l'inversion de formes d'ondes complètes Monte-Carlo et Molero & Iturrarán-Viveros (2013) implémentent une simulation sismique 2D anisotrope à l'aide de OpenCL.

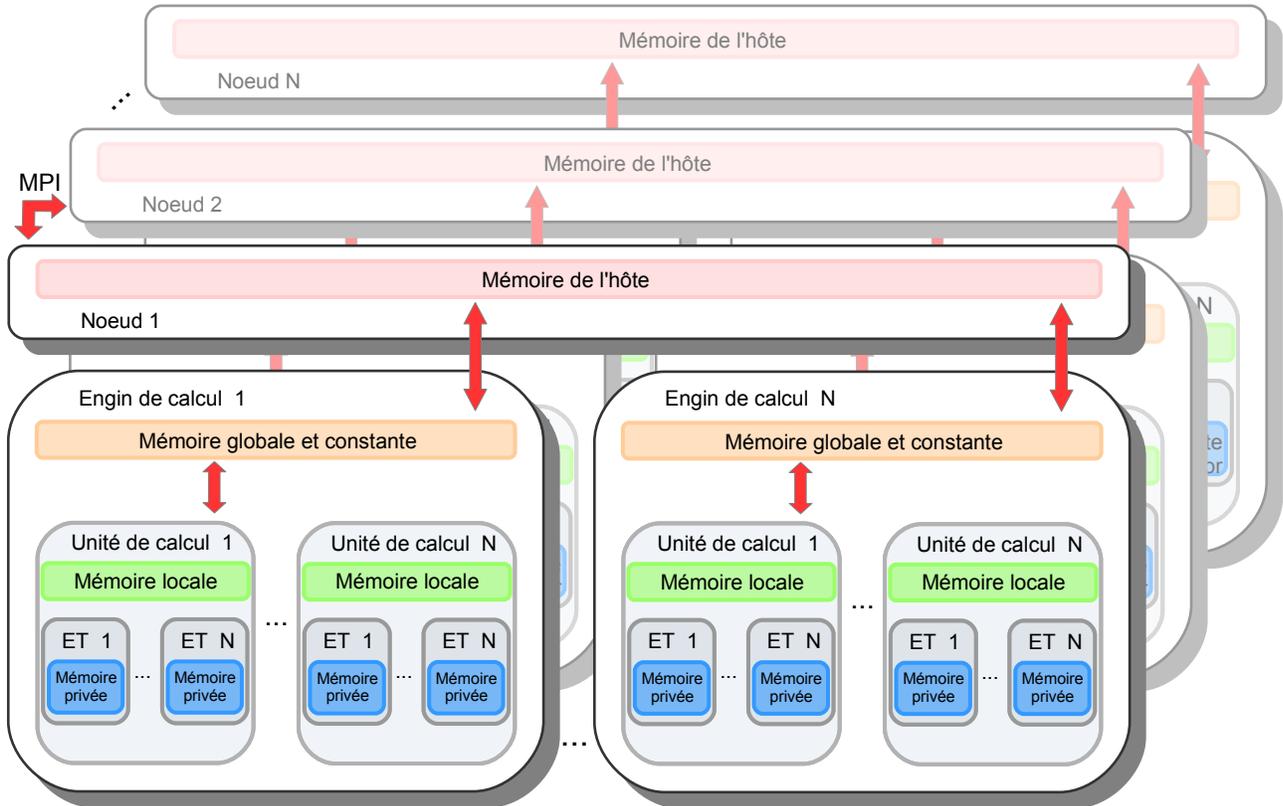


Figure 4.4 – Schéma général d'un processeur parallèle adopté par OpenCL et sa généralisation avec l'interface MPI.

4.4 Le langage OpenCL

Il va sans dire que la portabilité de OpenCL est un avantage indéniable pour la programmation visant des plateformes hétérogènes comportant plusieurs types de processeurs. En effet, écrire un code spécifique dans un langage propre à chaque architecture est un long processus qui peut s'avérer contre-productif. OpenCL permet au même code d'être exécuté sur un ou une combinaison de processeurs d'architectures différentes contenues dans un même ordinateur. Cette portabilité est la principale force du standard. Contrairement à MPI, OpenCL ne permet pas de synchroniser les calculs entre plusieurs ordinateurs. Toutefois, MPI et OpenCL peuvent être utilisés conjointement pour créer un environnement unifiant tous les engins de calculs hétérogènes d'un réseau d'ordinateurs. Bref, pour profiter des différents niveaux de parallélisme d'une grappe de calcul hétérogène, le code sismique qui sera décrit aux sections suivantes de ce chapitre utilise une approche hybride OpenCL/MPI. Avant tout, quelques notions essentielles sur OpenCL sont discutées dans cette section. Le lecteur est référé à Kaeli *et al.* (2015) pour une revue exhaustive du sujet.

OpenCL repose sur une abstraction de l'architecture d'une plateforme de calcul hétérogène comportant plusieurs processeurs parallèles. Ce modèle général est présenté à la Figure 4.4. Au niveau hiérarchique le plus élevé, il y a l'hôte, c'est-à-dire le programme géré par le système d'exploitation sur le CPU. L'hôte permet de gérer les différents engins de calculs accessibles localement; il envoie les instructions aux différents engins de calcul et gère la communication des données entre eux-ci. En quelque sorte, l'hôte est une unité de contrôle virtuelle de l'architecture de von Neumann et les engins de calculs sont des unités arithmétiques virtuelles. Un engin de calcul peut concrètement consister en un CPU, un GPU, un FPGA, etc. Chaque engin de calcul peut comprendre plusieurs unités de calcul, c'est-à-dire des coeurs pour des GPUs ou des CPUs. Enfin chaque coeur peut comporter plusieurs éléments de traitements ou fils d'exécution. Ainsi, tout comme dans le modèle d'une grappe de calcul présenté à la section précédente, OpenCL comporte trois niveaux de parallélisme: le niveau des engins de calculs, le niveau de l'unité de calcul et le niveau de l'élément de traitement.

De même, plusieurs niveaux de mémoire existent: la mémoire de l'hôte, qui consiste en la mémoire accessible au système d'exploitation. À l'intérieur d'un engin de calcul, il y a une mémoire globale, accessible par toutes les unités de calculs. Chaque unité de calcul comprend une mémoire locale, partagée par les éléments de traitement. Finalement, chaque élément de traitement possède sa propre mémoire, appelée mémoire privée. Ces niveaux de mémoires ont des performances et des capacités qui varient selon une architecture de processeur à l'autre. En général, le volume de mémoire décroît de la mémoire globale à la mémoire privée et croît en termes de rapidité d'accès.

Ce modèle abstrait schématise la plupart des architectures de processeurs existants. Cependant, l'optimisation d'un code OpenCL varie grandement d'un algorithme à l'autre et doit prendre en compte les particularités d'une architecture donnée. Dans ce travail, une attention particulière a été portée à l'optimisation des performances sur les GPU, ce type de processeur étant le plus efficace pour les calculs par différences finies. Michéa & Komatitsch (2010) décrivent en détail les concepts de programmation GPU appliqués à la modélisation sismique par différences finies. En particulier, les éléments de traitement dans un GPU Nvidia sont regroupés en groupe de 32, ce qui est dénommé par un *warp* en langage Nvidia, ou un groupe de travail en langage OpenCL. À l'intérieur d'un groupe de travail, les éléments de traitement peuvent être synchronisés et partagent la mémoire locale. Chaque unité de traitement doit effectuer les mêmes instructions, sur différentes adresses de mémoires. Une attention particulière doit être portée à l'accès de la mémoire à l'intérieur d'un *warp*. En effet, l'accès à la mémoire globale peut se faire de façon parallèle ou en série selon le

patron d'accès. Ainsi, lorsque chacun des éléments d'un *warp* accède à une adresse de mémoire adjacente, l'écriture ou la lecture se fait simultanément pour tous les éléments de traitement. Dans le cas contraire, plusieurs accès à la mémoire sont requis. Comme l'accès à la mémoire globale prend plusieurs cycles, la réduction du nombre d'accès est primordiale pour obtenir de bonne performance avec les GPUs. De plus, en différences finies, le nombre d'instructions par cycles d'écritures/lectures de mémoire est très bas, augmentant l'importance de limiter l'accès à ce niveau de mémoire. De même, l'utilisation de la mémoire locale, qui est beaucoup plus rapide que la mémoire globale, permet de diminuer le nombre de lectures et d'écritures. Bref, le patron d'accès à la mémoire est un des aspects principaux qui doit être optimisé sur GPU avec OpenCL.

En pratique, un programme basé sur OpenCL se déroule comme suit. Premièrement, des commandes sont envoyées à l'hôte afin de détecter les engins de calculs accessibles et de les connecter dans un contexte de calcul unique. Une fois ce contexte établi, des variables de mémoires sont créées afin de pouvoir transférer des éléments de mémoire entre l'hôte et les différents engins de calcul. Ensuite, les *kernels* sont chargés en mémoire et compilés en temps réel pour chaque engin de calcul. Les *kernels* sont des routines écrites en C qui comportent les instructions devant être effectuées par les engins de calcul. Par après, la partie principale du programme est exécutée, dans laquelle une série d'appels au *kernels* et de transferts de mémoire ont lieu. Ces appels sont gérés par l'hôte à l'aide d'un système de queue de commande. Finalement, les variables de mémoires sont libérées avant la fin du programme.

4.5 Implémentation parallèle avec MPI et OpenCL

Afin de tirer avantage du potentiel élevé de la parallélisation des différences finies pour la modélisation sismique directe et adjointe, il faut être en mesure de profiter des différents niveaux de parallélisme qu'offrent les grappes de calcul hétérogène. Le logiciel que j'ai conçu avec l'interface MPI permet de gérer différents noeuds de calculs et, grâce à OpenCL, il permet aussi d'utiliser différentes architectures de processeur. Le code proposé ici, dénommé SeiCL, est distribué sous licence libre et publiquement offert sur la plateforme *GitHub* (<https://github.com/gfabieno/SeisCL>).

La routine générale du code est présentée à l'Algorithme 7. Elle débute par la lecture des fichiers d'entrée, qui sont sous le format *Hierarchical Data Format*, version 5 ou HDF5 (HDF Group, 2016).

Ce format de fichier a été choisi pour son accès parallèle efficace, la compression automatique des variables, sa compatibilité avec le format `.mat` v7.3 de Matlab et les nombreuses bibliothèques de lecture/écriture disponible en plusieurs langages de programmation, dont Python. Ensuite, l’environnement MPI est initialisé et tous les paramètres d’inversion sont synchronisés entre les différents noeuds disponibles. Chaque noeud est assigné à un processus MPI différent. Chacun des noeuds initialise ensuite son environnement OpenCL. En particulier, un contexte unique est créé pour tous les engins de calcul disponible d’un certain type. Par défaut, SeisCL choisit les GPUs disponibles. L’initialisation OpenCL comprend aussi la compilation des *kernels*, la création des variables mémoires OpenCL et le transfert des paramètres de la simulation. Une fois ce contexte MPI/OpenCL créé, la modélisation sismique et le calcul du gradient de la fonction coût peuvent se dérouler selon plusieurs niveaux de parallélisme.

Algorithme 7 SeisCL: Routine générale

```

1: Lire paramètres des fichiers HDF5
2: Initialiser MPI
3: Initialiser OpenCL
4: for all groupes  $\in$  MPI do
5:   for all tirs  $\in$  groupe do
6:     for all noeuds  $\in$  groupe do
7:       for all engins  $\in$  noeud do
8:         Initialiser grilles sismiques
9:         Exécuter PROPAGER(tir)
10:        Calculer résidus
11:        Exécuter PROPAGER(résidus)
12:        Calculer gradient
13: Accumuler gradient des groupes

```

4.5.1 Parallélisme des sources

Au niveau le plus élevé d’une grappe de calcul, le parallélisme des sources est implanté. Ainsi, les noeuds de la grappe de calcul sont divisés en groupes de même taille. Le parallélisme des sources est implémenté en assignant un sous-ensemble des sources à chacun des groupes. Chacun des groupes est indépendant des autres et a comme tâche d’obtenir la composante du gradient de son sous-ensemble de sources. En d’autres mots, les boucles des lignes 4 et 5 de l’Algorithme 7 se font de façon simultanée. Ce parallélisme de tâches ne requiert aucune communication entre les groupes de noeuds et peut être effectué sur un réseau relativement lent. Par exemple, il est possible d’utiliser

un réseau local, ou même une grille informatique (*Grid Computing*), où chaque ordinateur n'est connecté que par une connexion internet. La seule communication requise ne survient qu'une fois que tous les groupes ont complété le calcul du gradient des sources qui leur ont été assignées: il faut alors regrouper tous les gradients partiels et sommer le résultat de chacun des groupes pour obtenir le gradient donné par l'ensemble complet de sources.

4.5.2 Parallélisme des mises à jour

Le parallélisme de mise à jour est implémenté par la technique de la décomposition de domaine (Mattson *et al.*, 2004). Cette subdivision est représentée par les boucles des lignes 6 et 7 de l'Algorithme 7. La décomposition de domaine consiste subdiviser la grille de différences finies en sous-grilles qui seront chacune mises à jour par des engins de calcul différents. La grille est ici subdivisée selon une seule dimension, soit l'axe des x . La décomposition de domaine et les communications requises sont illustrées à la Figure 4.5. Comme mentionné auparavant, la mise à jour d'un point de la grille d'une variable sismique nécessite les valeurs des points adjacents au pas précédent. Ainsi, à chaque pas de temps, certains éléments des sous-grilles doivent être synchronisés entre les différentes sous-grilles. Pour un schéma de différences finies d'ordre $2N_{DF}$, les N_{DF} premières et N_{DF} dernières rangées en x d'une sous-grille doivent être communiquées à la sous-grille précédente et suivante respectivement. Si deux sous-grilles résident sur des engins de calcul du même noeud, cette communication est gérée par OpenCL, tandis que si les deux sous-grilles sont sur des noeuds différents, le protocole MPI est utilisé.

Cette communication à chaque pas de temps nécessite des connexions très rapides entre les engins de calcul ainsi qu'entre les différents noeuds d'une grappe de calcul afin de maintenir une bonne scalabilité. Par exemple, Bohlen (2002) observe une scalabilité supra-linéaire jusqu'à 350 noeuds sur une grappe avec un réseau de 450 Mo/s. Pour un réseau de 100 Mo/s, cette scalabilité linéaire n'est observée que pour un maximum de 12 noeuds. En utilisant des engins de calcul comme des GPU, les communications doivent non seulement passer par le réseau, mais aussi par le port PCIe entre le GPU et l'hôte. Ceci réduit dramatiquement les gains du parallélisme. Par exemple, Okamoto *et al.* (2011) obtiennent une scalabilité entre N et $N^{2/3}$.

Afin de réduire l'impact du temps des communications lors de la décomposition de domaine, le calcul de mise à jour et les communications se font concurremment. Ceci est possible en lançant

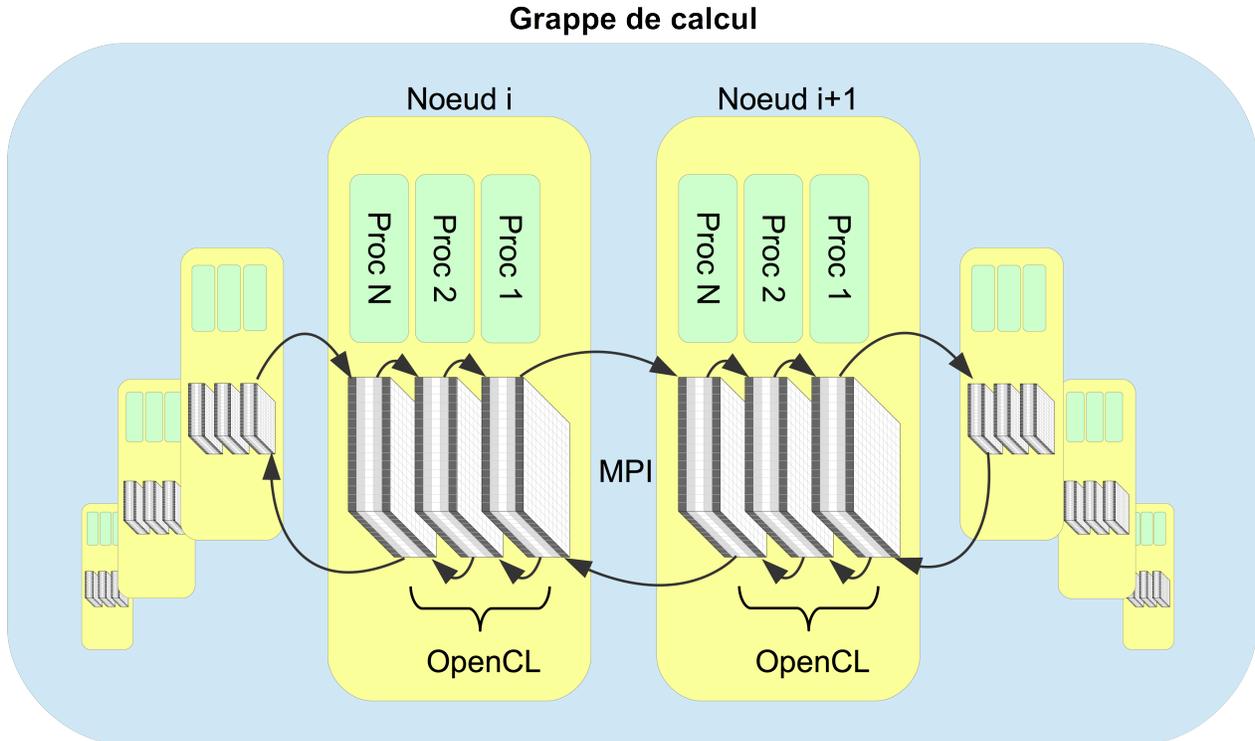


Figure 4.5 – Décomposition de domaine et protocoles de communication utilisés.

deux *kernels* différents sur deux files d'attente OpenCL: un *kernel* de mise à jour des points internes de la sous-grille qui ne requiert aucune communication et un *kernel* pour les points des extrémités en x de la sous-grille qui nécessitent des communications. À la Figure 4.5, ces deux sections correspondent respectivement aux sections blanches et en gris pâle. Cette stratégie est aussi résumée à l'Algorithme 8, montrant l'algorithme de simulation sismique par différences finies. Pour des sous-domaines assez-grands, les points internes de la grille sont beaucoup plus nombreux que les points du pourtour nécessitant des communications. Le *kernel* des points externes s'exécutera donc plus rapidement, permettant de communiquer les valeurs mises à jour en attendant que le *kernel* des points internes termine son calcul. Ainsi, les lignes 4 et 5 et les lignes 9 et 10 de l'Algorithme 8 sont exécutées simultanément, permettant de réduire la latence introduite par les communications entre les sous-domaines.

4.5.3 Parallélisme intra-processeur OpenCL

Le niveau le plus fin de parallélisme est le parallélisme à l'intérieur même de l'engin de calcul. En effet, un GPU contient des milliers d'unités de traitement pouvant simultanément faire des calculs.

Algorithme 8 Routine de propagation

```

1: procédure PROPAGER(source)
2:   while  $t < N_t$  do
3:     Exécuter KERNEL_ACTUALISER( $v_i$ ) sur pourtour
4:     Transférer  $v_i$  entre engins et noeuds
5:     Exécuter KERNEL_ACTUALISER( $v_i$ ) sur intérieur
6:     Point de synchronisation
7:     Injecter source  $v_i$ 
8:     Exécuter KERNEL_ACTUALISER( $\sigma_I, r_I$ ) sur pourtour
9:     Transférer  $\sigma_I$  entre engins et noeuds
10:    Exécuter KERNEL_ACTUALISER( $\sigma_I, r_I$ ) sur intérieur
11:    Point de synchronisation
12:    Injecter source  $\sigma_I$ 
13:    Incrémenter  $t$ 

```

Ce niveau est directement géré par OpenCL selon le modèle d'instruction unique, données multiples. Ainsi, le programmeur définit une grille de calcul à n dimensions. Dans le cas de la modélisation sismique 3D, cette grille est tridimensionnelle, de dimension égale à la sous-grille assignée à l'engin de calcul. Les *kernels* d'actualisation des variables sismiques regroupent les instructions pour mettre à jour un point de la grille. Ce *kernel* est appliqué directement par OpenCL à chacun des points de la grille, sans que le programmeur n'ait à formuler une boucle quelconque.

Comme discuté aux sections précédentes, les GPUs regroupent leurs fils d'exécution en groupes. Ainsi, OpenCL reproduit ce comportement en exécutant le *kernel* en groupes de travail locaux d'un nombre de dimensions égal au nombre de dimension de la grille globale de travail. La taille du groupe de travail local doit être compatible avec l'engin de calcul spécifié. Afin de maximiser l'utilisation des ressources des GPU, ce regroupement devrait être un multiple de la taille d'un *warp*, soit 32, et ne pas dépasser le nombre de fils à l'intérieur d'un coeur. Tous les fils d'un groupe de travail local sont situés sur le même coeur du GPU et pourront donc accéder à la mémoire locale. L'utilisation de la mémoire locale est la clé afin de réduire le temps d'accès à la mémoire globale qui prend plusieurs cycles lors de la lecture ou de l'écriture. En effet, tout comme les communications entre les noeuds d'une grappe de calcul diminuent l'accélération maximale pouvant être atteinte par le calcul parallèle, l'accès à la mémoire est le facteur limitant pour le calcul GPU. En effet, la lecture et l'écriture en mémoire prennent plus de temps que les opérations arithmétiques dans les *kernels* de différences finies.

L'utilisation de la mémoire locale est illustrée au *kernel* de la mise à jour des contraintes et des variables de mémoire de l'Algorithme 9. Comme le calcul des dérivées spatiales par différences finies requiert les éléments adjacents de la grille, il est possible de partager l'accès à ces éléments entre les fils d'un même groupe de travail local. Ainsi, pour chaque composante des vitesses, chacun des éléments du groupe de travail local lit à partir de la mémoire globale le point de la grille pour lequel le fil est responsable de la mise à jour. Ces données sont écrites dans la mémoire locale. Les éléments du pourtour du groupe de travail doivent aussi lire les éléments adjacents aux positions de la grille mise à jour par le groupe de travail local. Ceci correspond à la ligne 4 de l'Algorithme 9. Le patron des éléments de la grille dans la mémoire locale pour un schéma en différences finies d'ordre 2 est représenté à la Figure 4.6. Les éléments en blanc sont les positions de la grille qui sont mises à jour par les fils du groupe de travail local. Les éléments en gris foncés sont les éléments qui doivent être lus pour calculer les dérivées spatiales.

Une fois que les vitesses sismiques sont contenues dans la mémoire locale, chaque fil d'exécution peut calculer les dérivées spatiales à la position de la grille qu'il doit mettre à jour (ligne 5 l'Algorithme 9). Ce calcul est très rapide étant donné qu'il ne requiert aucun accès à la mémoire globale. Finalement, les contraintes et les variables de mémoire peuvent être mises à jour directement dans la mémoire globale (lignes 7, 8, 9). À noter que la mise à jour des vitesses se fait de façon similaire et utilise la même approche de lecture en la mémoire locale.

Algorithme 9 *Kernel* d'actualisation des contraintes et des variables de mémoire

```

1: procedure KERNEL_ACTUALISER( $\sigma, \mathbf{r}$ )
2:   for all domaines_locaux  $\in$  sous_grille do
3:     for all points  $\in$  domaine_local do
4:       Lire  $\mathbf{v}$  de mémoire globale à mémoire locale
5:       Calculer  $D_j^\pm \mathbf{v}$  avec mémoire locale (équation (2.44))
6:       Exécuter CPML( $D_j^\pm \mathbf{v}$ ) (équation (2.58))
7:       Mise à jour partielle  $\sigma$  (équation (2.50))
8:       Mise à jour  $\mathbf{r}$  (équation (2.51))
9:       Mise à jour finale  $\sigma$  (équation (2.50))

```

4.5.4 Calcul du gradient

Un des défis numériques de la méthode adjointe est le calcul de la corrélation croisée entre les champs directs et adjoints (équation (3.36)). Ce calcul implique la multiplication du champ directe et adjoint à chacun des pas de temps. Cependant, le champ adjoint est propagé du temps final au

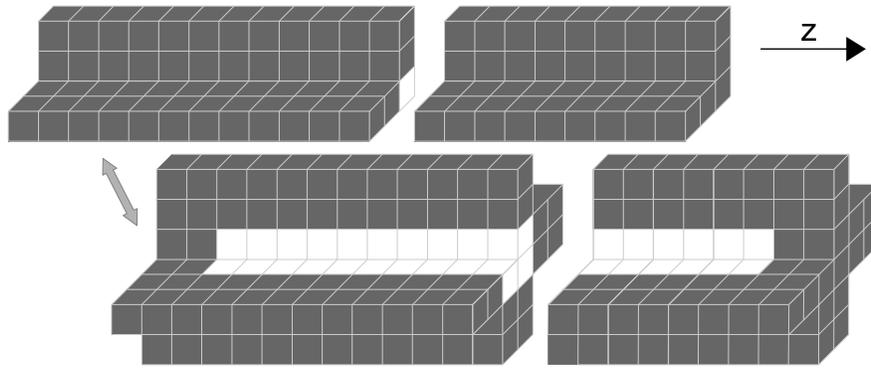


Figure 4.6 – Schéma des cellules de la grille transférées de la mémoire globale à la mémoire locale à l'intérieur d'un groupe de travail local. Les cellules en blanc sont mises à jour tandis que les cellules en gris ne sont utilisées que pour les calculs des dérivées spatiales.

temps initial, tandis que le champ direct est propagé du temps initial au temps final. Pour que ces deux champs soient accessibles au même pas de temps, plusieurs stratégies ont été proposées. Dussaud *et al.* (2008) et Nguyen & McMechan (2015) en font une bonne revue. Quatre stratégies sont particulièrement utilisées:

- **Sauvegarde en mémoire** La stratégie la plus évidente est de sauvegarder en mémoire le champ sismique direct entier à chaque pas de temps ou à un sous-ensemble des pas de temps. Lorsque le champ adjoint est propagé du temps final au temps initial, le champ direct est lu de la mémoire, et le produit scalaire peut être évalué itérativement, pas de temps à pas de temps.
- **Propagation inverse** La stratégie de la propagation inverse (Clapp, 2008; Yang *et al.*, 2014) consiste à sauvegarder chaque pas de temps du pourtour de la grille sismique qui n'est pas dans la zone absorbante. La largeur de bande du pourtour devant être sauvegardée est égale à la moitié de l'ordre du schéma de différences finies. Le champ direct est reconstitué lors de la propagation du champ adjoint en le propageant à temps inverse à partir de l'état final. À chaque pas de temps, les valeurs sauvegardées du pourtour sont réinjectées. Étant donné que le champ direct et adjoint sont propagés du temps final au temps initial, la corrélation croisée peut être calculée itérativement, pas de temps à pas de temps.
- **Sélection de fréquences** La corrélation croisée peut être calculée dans le domaine des fréquences (équation (3.37)). Lors de la mise à jour des champs direct et adjoint, quelques fréquences discrètes peuvent être calculées par la transformée de Fourier discrète et sauvegardées en mémoire. Suite à la propagation du champ adjoint, la corrélation croisée peut être approximée par la multiplication des fréquences sélectionnées. Cette approche temps-

fréquence a été adoptée par Sirgue *et al.* (2008), entre autres. Une méthode alternative pour calculer un faible nombre de composantes fréquentielles a été proposée par Furse (2000), mais n’a pas été testée dans ce travail.

- **Points de contrôle optimaux** La méthode des points de contrôle optimaux proposée par Griewank (1992) et Griewank & Walther (2000) et appliquée par (Symes, 2007) consiste à sauvegarder en mémoire le champ sismique direct à un certain nombre de pas de temps nommés points de contrôle. Afin de calculer le produit scalaire, le champ direct est reconstitué lors de la propagation adjointe en propageant le champ directe à partir du point de contrôle du temps le plus rapproché. Pour un nombre fixe de points de contrôle, une distribution temporelle optimale qui minimise le nombre de mises à jour supplémentaires du champ direct peut être calculée. Pour cette distribution optimale, le nombre de points de contrôle et le nombre de mises à jour supplémentaires varient de façon logarithmique avec le nombre total de pas de temps. Des améliorations de la méthode ont été proposées par Anderson *et al.* (2012) et Komatitsch *et al.* (2016) dans le cas viscoélastique.

Parmi ces stratégies, la sauvegarde en mémoire est irréaliste, car elle requiert une énorme quantité de mémoire pour des problèmes de moyennes et grandes tailles. En 3D, un nombre d’éléments de l’ordre de $\mathcal{O}(N_t N^3)$ doit être sauvegardé, ce qui dépasse rapidement les capacités en mémoire actuelles. Notons que l’utilisation de la compression et du sous-échantillonnage peuvent réduire le coût de cette approche (Boehm *et al.*, 2016; Sun & Fu, 2013). La stratégie de la propagation arrière nécessite beaucoup moins de mémoire: un nombre de l’ordre de $\mathcal{O}(N_t N^2)$ éléments doit être sauvegardé en 3D. Cependant, comme le champ direct doit être propagé une deuxième fois, cette stratégie nécessite un plus grand nombre de calculs, soit trois modélisations sismiques au lieu de deux. De plus, la propagation arrière n’est pas compatible avec l’équation viscoélastique. En effet, en inversant le temps, c’est-à-dire $\Delta t \rightarrow -\Delta t$, les équations de mise à jour des variables de mémoire (équation (2.51)) deviennent instables.

Ainsi, pour l’équation viscoélastique, seules les stratégies en fréquence et de points de contrôle optimaux sont envisageables. Les ressources en mémoire de la stratégie de la sélection de fréquences varient selon $\mathcal{O}(N_f N^3)$, où N_f est le nombre de fréquences sélectionnées. Cependant, il est possible de choisir un nombre restreint de fréquences (Sirgue & Pratt, 2004), mitigeant le coût en mémoire de cette stratégie. La méthode des points de contrôle optimaux nécessite la sauvegarde de $\mathcal{O}(N_c N^3)$ éléments, où N_c est le nombre de points de contrôle. Étant donné la relation logarithmique entre le

nombre de pas de temps de la simulation, le nombre de points de contrôle et le nombre de mises à jour additionnelles, la mémoire requise reste la plupart du temps raisonnable. Par exemple, pour 10 000 pas de temps, en utilisant 30 points de contrôle, le coût de la méthode n'est que de 3.4 fois celui d'une seule modélisation directe.

Dans ce travail, la stratégie de la propagation arrière a été implémentée pour le cas élastique et la stratégie de la sélection en fréquence a été implémentée pour les cas élastiques et viscoélastiques. L'implémentation de la méthode des points de contrôle optimaux ou de la méthode hybride points de contrôle/propagation arrière (Yang *et al.*, 2016a) sera l'objet de travaux futurs.

Le calcul du gradient selon la stratégie de la propagation arrière est schématisé à l'Algorithme 10. Par désir de simplicité, cet exemple n'est écrit que pour un seul engin de calcul et donc aucune communication n'est requise. De plus, cet algorithme est appliqué à l'équation élastique. Ainsi, à chaque pas de temps lors de la propagation du champ direct, les valeurs des variables sismiques sur les pourtours du modèle sont sauvegardées. Comme la quantité de mémoire dans les GPUs est assez limitée, ces valeurs sont transférées à l'hôte à chaque pas de temps. Cette communication est simultanée aux calculs de mise à jour, ce qui est possible avec OpenCL en utilisant une deuxième queue de commandes. Après avoir calculé les résidus (ligne 7), ceux-ci sont propagés par la routine adjointe et le champ direct est propagé à rebours en redéfinissant le pas de temps comme négatif $\Delta t \rightarrow -\Delta t$. De plus, à chaque pas de temps, les valeurs du pourtour du champ direct sont réinjectées. Ainsi, lors de la propagation arrière, les champs directs et adjoints sont accessibles au même pas de temps et la composante du gradient est additionnée à chaque pas de temps (ligne 13).

Algorithme 10 Stratégie de la propagation arrière

```

1: while  $t < N_t$  do
2:   Exécuter KERNEL_ACTUALISER( $v_i$ )
3:   Sauvegarder  $v_i$  sur pourtour
4:   Exécuter KERNEL_ACTUALISER( $\sigma_i$ )
5:   Sauvegarder  $\sigma_i$  sur pourtour
6:   Incrémenter  $t$ 
7:   Calculer résidus
8:   while  $t > 0$  do
9:     Exécuter KERNEL_ACTUALISER( $v_i, \bar{v}_i$ )
10:    Injecter  $v_i$  sur pourtour
11:    Exécuter KERNEL_ACTUALISER( $\sigma_i, \bar{\sigma}_i$ )
12:    Injecter  $\sigma_i$  sur pourtour
13:    Calcul partiel gradients
14:    Décrémenter  $t$ 

```

La stratégie de sélection de fréquences est décrite à l’Algorithme 11. Lors de la propagation du champ direct, la transformée de Fourier discrète des variables sismiques est calculée de façon incrémentielle pour les fréquences sélectionnées. Lors de la propagation des résidus, la transformée de Fourier discrète des variables adjointes est obtenue de la même façon. Après la propagation adjointe, les composantes fréquentielles des champs adjoints et directs sont multipliées pour obtenir le gradient. À noter qu’en raison de la quantité de mémoire limitée des GPU, le nombre de fréquences pouvant être calculé est limité.

Algorithme 11 Stratégie de la sélection des fréquences

```

1: while  $t < N_t$  do
2:   Exécuter KERNEL_ACTUALISER( $v_i$ )
3:   Exécuter KERNEL_ACTUALISER( $\sigma_I, r_I$ )
4:   Exécuter DFT( $v_i, \sigma_I, r_I$ ) pour  $freqs$ 
5:   Incrémenter  $t$ 
6:   Calculer résidus
7:   while  $t > 0$  do
8:     Exécuter KERNEL_ACTUALISER( $\tilde{v}_i$ )
9:     Exécuter KERNEL_ACTUALISER( $\tilde{\sigma}_i, \tilde{r}_I$ )
10:    Exécuter DFT( $\tilde{v}_i, \tilde{\sigma}_i, \tilde{r}_I$ ) pour  $freqs$ 
11:    Décrémenter  $t$ 
12:   Calculer gradients

```

4.6 Validation du code

Il est important de pouvoir tester l’exactitude du code décrit à la section précédente. En particulier, les vitesses de particules obtenues par différences finies doivent concorder avec les solutions connues de l’équation d’onde viscoélastique isotrope. Ceci est particulièrement important en vue de l’éventuelle application du code d’inversion. De plus, le gradient obtenu par la méthode adjointe doit être exacte pour que les algorithmes d’inversion puissent converger au minimum de la fonction coût. Cette section présente deux expériences pour vérifier ces deux aspects.

4.6.1 Validation de la modélisation sismique

Afin de tester la validité de la solution de la fonction d’onde viscoélastique, les résultats du code en différences finies sont comparés à la solution analytique donnée à l’équation (2.34). La solution

viscoélastique est obtenue en remplaçant les modules de cisaillement et des ondes P réels par des paramètres complexes, données à l'équation (2.9). Le modèle homogène utilisé est: $V_p = 3500$ m/s, $V_s = 2000$ m/s, $\rho = 2000 \text{ kg/m}^3$, avec quatre niveaux d'atténuation différents $\tau = [0, 0.01, 0.2, 0.4]$, c'est-à-dire $Q \approx [\infty, 200, 10, 5]$. Un seul mécanisme d'atténuation est utilisé, avec $2\pi/\tau\sigma_l = 40$ Hz. La source est une impulsion de Ricker (équation (2.55)), d'une fréquence centrale de 40 Hz, orientée dans la direction des z . Un schéma spatial de différences finies d'ordre 4 est utilisé. Un pas spatial de 6 m (8.33 points pas longueur d'onde) assure que la dispersion numérique reste minimale (équation (2.42)) et un pas temporel de 0.5 ms assure la stabilité de la simulation (équation (2.43)).

La Figure 4.7 montre la composante en x des vitesses de particules pour un déport de 132 m dans la direction des z . Pour le cas élastique ($\tau = 0$), la solution analytique est pratiquement reproduite parfaitement par SeisCL. Pour des niveaux d'atténuation plus importants, une erreur est introduite. Cette erreur grandit avec le niveau d'atténuation, et pour $\tau = 0.4$ ($Q = 5$), l'erreur devient évidente pour ce déport. Ceci semble être le défaut de la résolution par différences finies dans le domaine du temps (voir Gosselin-Cliche & Giroux (2014)). De plus, pour des niveaux d'atténuation plus raisonnables, l'erreur semble plutôt limitée et ne devrait pas avoir trop d'impact lors de l'inversion. Le manque de précision pourrait devenir problématique dans le cas d'un milieu fortement atténuant ou pour de longues distances de propagation.

4.6.2 Validation du gradient de la fonction coût

Le second test de validation vise à vérifier la précision du gradient calculé par la méthode adjointe. Pour ce faire, un levé synthétique 2D entre forages est simulé. Celui-ci est illustré à la Figure 4.8. La séparation entre le forage des sources (gauche) et des récepteurs (droit) est 250 m. L'intervalle des sources est de 60 m et celui des récepteurs deux composantes est de 12 m. Un modèle homogène avec les mêmes paramètres que le test précédent est utilisé, avec $\tau = 0.2$. À ce modèle, des perturbations circulaires de 5 % pour chacun des paramètres sont introduites à différentes positions. Les paramètres sismiques peuvent être fortement couplés, particulièrement les vitesses et les facteurs de qualité (Kamei & Pratt, 2013). Pour cette raison, le gradient pour un paramètre perturbé est obtenu en l'absence de perturbation des autres paramètres. Par exemple, le gradient pour V_p est obtenu pour une anomalie de V_p et des modèles constants pour V_s , ρ , τ_p et τ_s . Ceci empêche toute interférence sur le gradient pouvant être causée à la présence d'une perturbation

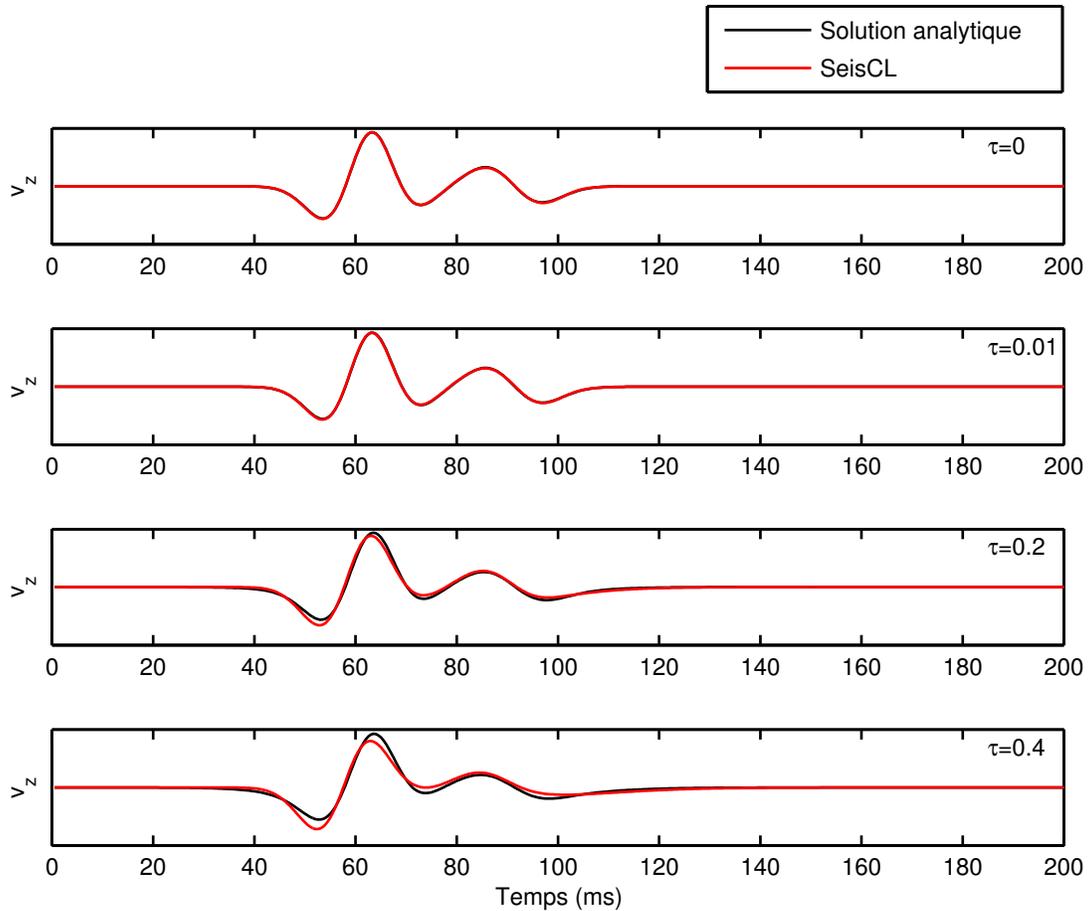


Figure 4.7 – Comparaison entre la solution analytique et les résultats en différences finies obtenus avec SeisCL pour quatre niveaux d’atténuation.

d’un autre paramètre. Comme aucune solution analytique n’existe pour le gradient de la fonction coût, le gradient obtenu par la méthode adjointe est comparé au gradient obtenu par différences finies. Le gradient par différences finies est obtenu en perturbant séquentiellement chaque position de la grille d’un paramètre par 2 %. Le gradient obtenu par la méthode adjointe est calculé par la stratégie en fréquence, avec toutes les fréquences de la transformée de Fourier discrète entre 0 et 125 Hz.

La Figure 4.8 montre la comparaison entre les deux méthodes de calcul du gradient. La première rangée présente la position des perturbations, la seconde rangée contient la direction de plus forte descente (moins le gradient) calculée par différences finies et la troisième rangée présente la direction de plus forte descente obtenue par la méthode adjointe. À noter que ces directions sont normalisées dans cette figure. Pour tous les paramètres, les deux méthodes de calcul du gradient sont en excellent accord. Ainsi, les gradients obtenus par la méthode adjointe pourraient être utilisés dans

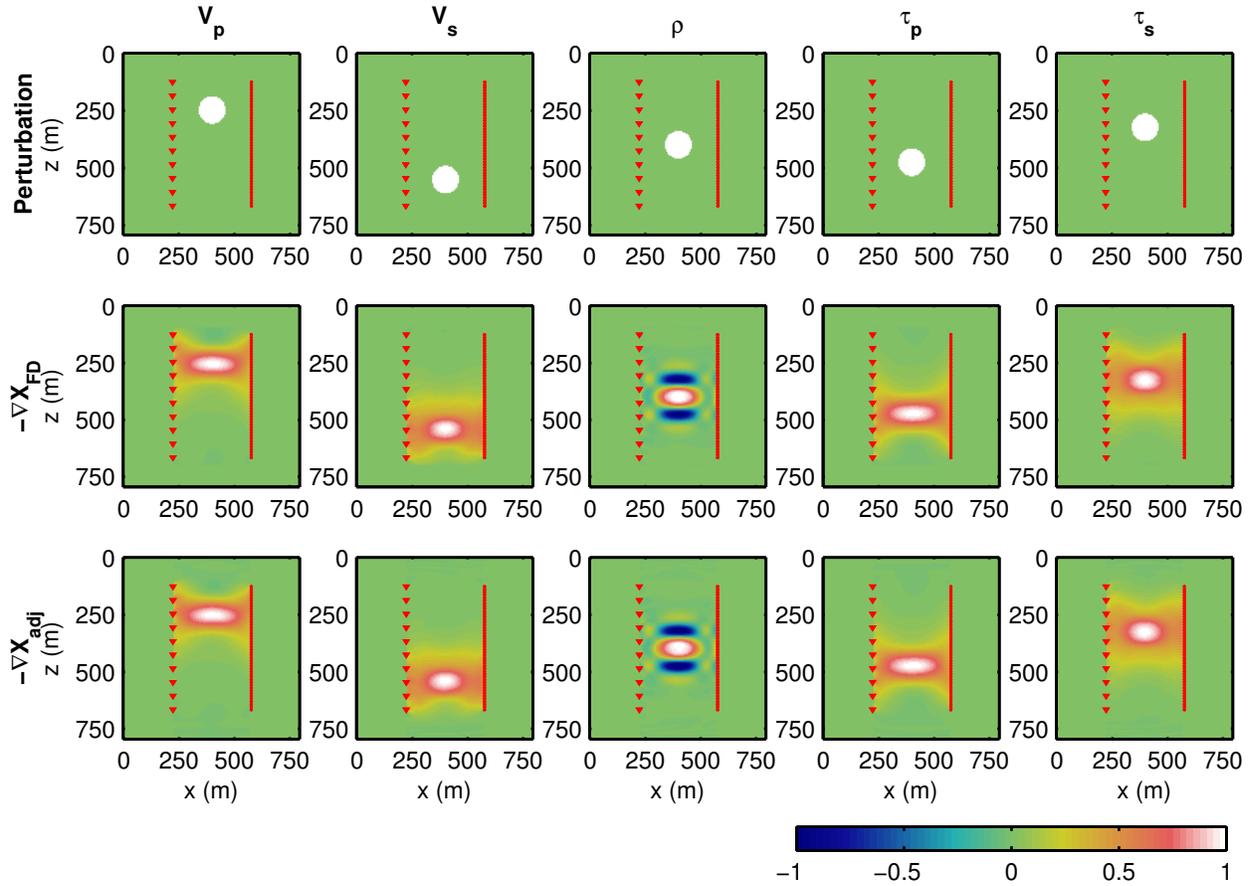


Figure 4.8 – Comparaison du gradient obtenu par différences finies et par le modèle adjoint. Le sources et les receveurs sont représentés respectivement par les triangles et les cercles rouges.

n'importe quel algorithme d'optimisation basé sur le gradient. La méthode adjointe est cependant plus rapide de plusieurs ordres de grandeur. Ainsi, pour cette expérience-ci, la méthode adjointe requiert plusieurs minutes à compléter tandis que le calcul par différences finies requiert plus d'une journée.

4.7 Comparaison des performances

Les efforts de programmation supplémentaires à fournir pour construire un code parallèle seraient futiles sans un gain significatif en performance de calcul. Il est donc important de mesurer l'importance de ce gain. Dans cette optique, la mesure d'accélération est définie comme

$$S_{acc} = \frac{T_{référence}}{T_{SeisCL}}, \quad (4.5)$$

où T_{ref} et T_{SeisCL} sont les temps d'exécution d'une référence et du programme testé, respectivement. Différentes références seront utilisées selon le test spécifique. À moins de mention contraire, le CPU est un CPU double Intel Xeon E5-2680 v2 avec chacun 10 coeurs, le GPU est un Tesla K40 de Nvidia et le coprocesseur est le Xeon Phi 5110P.

4.7.1 Accélération selon le type d'engin de calcul

Ce test vise à mesurer l'accélération des engins de calcul utilisant OpenCL par rapport à une implémentation en série. Comme référence, les logiciels SOFI2D et SOFI3D, les implémentations 2D et 3D du code présenté par Bohlen (2002), sont utilisé sur un seul coeur. Ce code peut être comparé à SeisCL étant donné qu'ils partagent des algorithmes très semblables. Ce code est aussi représentatif des rendements qui peuvent être atteints avec un code en différences finies écrit dans le langage C99, l'un des des langages haut niveau les plus rapides pour les CPUs.

La Figure 4.9 présente les accélérations de différents engins de calcul pour des tailles de modèles différents en 2D et 3D. Les modèles sont de forme carrée et cubique respectivement, d'une longueur d'arrête N . L'accélération varie considérablement avec la taille du modèle. Les accélérations les plus élevées sont atteintes avec un GPU. Celles-ci se situent entre 50 et 80 en 3D et entre 30 et 75 en 2D. De plus, des accélérations significatives sont obtenues avec les CPUs, atteignant 35, ce qui est plus élevé que le nombre de coeurs (20). Ceci peut être expliqué par une meilleure utilisation de la mémoire cache de l'implémentation OpenCL: l'utilisation de la mémoire locale améliore l'accès à la mémoire cache en comparaison à l'implémentation en langage C de Bohlen (2002). L'implémentation 2D semble moins affectée par ce phénomène et la plage d'accélération semble plus normale entre 11 et 25. Le succès erratique de l'utilisation de la mémoire cache de l'implémentation en langage C est aussi la source de fortes variations en accélération observées à la Figure 4.9. Enfin, les accélérations obtenues des coprocesseurs Xeon Phi sont décevantes vis-à-vis de leurs capacités théoriques. Cependant, SeisCL a été optimisé pour les cartes graphiques, et non pour l'architecture des Xeon Phi. Même si ceci n'a pas été testé, il est possible qu'avec de faibles modifications au code, de meilleures performances puissent être atteintes avec ce type d'engin de calcul. Ceci montre cependant les limites des portabilités d'architecture d'OpenCL: pour obtenir de bonnes performances avec OpenCL, le code doit être optimisé pour l'architecture de l'engin de calcul. Cette optimisation peut être assez différente selon l'engin de calcul.

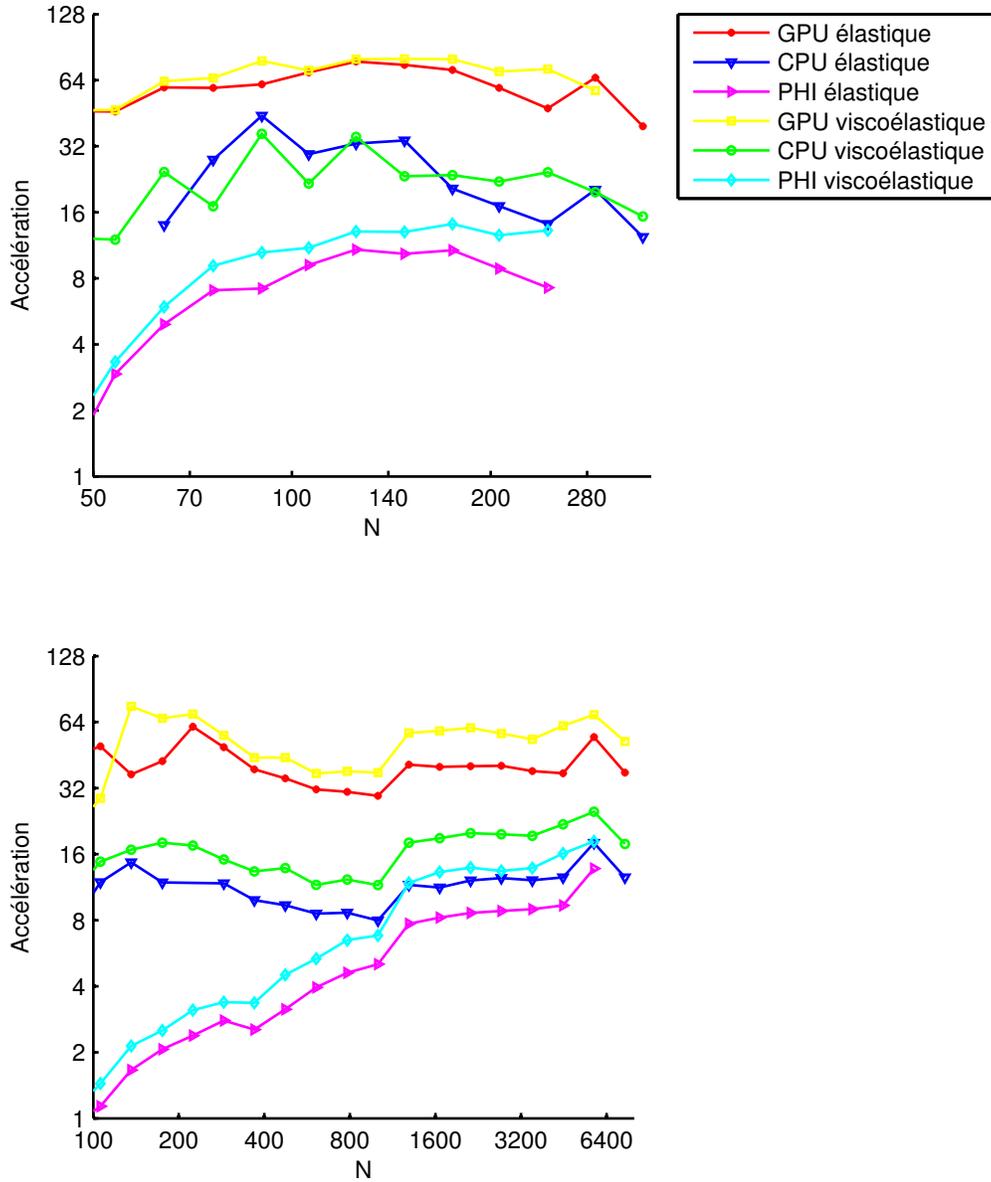


Figure 4.9 – Accélération obtenue avec différents types d'engins de calcul par rapport à une implémentation en série pour le cas 3D (haut) et 2D (bas).

4.7.2 Performance du calcul du gradient

Le test suivant a comme but d'évaluer la performance des deux stratégies de calcul du gradient implémentées dans ce travail: la sélection de fréquences et la propagation arrière. La référence est ici le temps requis pour effectuer une simulation du champ direct. La taille du modèle est de 1000x1000 en 2D et de 100x100x100 en 3D. Les résultats sont montrés à la Figure 4.10. Pour la stratégie de sélection de fréquence, le temps de calcul croît linéairement avec le nombre de fréquences. Ce

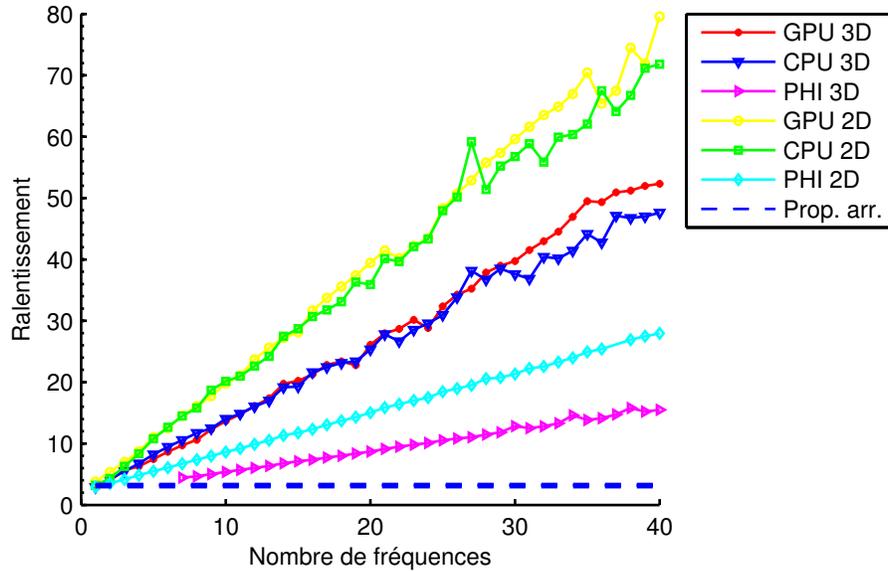


Figure 4.10 – Ralentiement mesuré pour le calcul du gradient par sélection de fréquences pour un nombre de fréquences croissant.

ralentiement croît plus rapidement en 3D qu'en 2D, ce qui peut être expliqué par le plus grand nombre de variables sismiques devant être transformées en 3D. De façon étonnante, le coût grandit moins vite pour le Xeon Phi que pour le CPU ou le GPU cette fois. Le coût de la stratégie de la sélection de fréquences n'est donc pas négligeable et n'est envisageable que pour un faible nombre de fréquences. Notons que le coût de la stratégie de la propagation arrière est environ 3 fois celui d'une modélisation, ce qui est attendu. Conséquemment, pour le cas élastique, la propagation arrière est beaucoup plus avantageuse que la sélection de fréquences, d'autant plus qu'elle permet de calculer le gradient pour toutes les fréquences à la fois. L'implémentation de la méthode des points de contrôle optimaux serait avantageuse dans le cas viscoélastique, surtout lorsqu'une large bande de fréquence est nécessaire lors de l'inversion.

4.7.3 Coût des schémas en différences finies d'ordre élevé

Pour ce test, la référence est le temps de simulation du schéma de différences finies d'ordre 2 sur chaque type de processeur. Les mêmes pas temporel et spatial sont utilisés pour tous les ordres. Comme le montre la Figure 4.11, pour les trois types d'engins de calcul, le ralentiement est assez faible et n'excède pas 1.5 pour l'ordre de 12, le plus élevé implémenté dans SeisCL. Les seules exceptions sont les ordres 10 et 12 sur le GPU, pour lesquels le ralentiement atteint 3. Jusqu'à

l'ordre 8, les performances du GPU sont cependant comparables aux autres types de processeurs. Ce coût plus élevé pour l'ordre 10 et 12 est expliqué par la taille limitée de la mémoire locale des GPU. En effet, pour ces ordres élevés, le patron de la mémoire locale utilisé pour le calcul des dérivées spatiales (Figure 4.6) excède les capacités du GPU. Dans ces circonstances, SeisCL désactive l'utilisation de la mémoire locale et calcule les dérivées spatiales directement à partir de la mémoire globale. Le ralentissement abrupt montre l'importance d'utiliser la mémoire locale. La raison du faible coût des ordres supérieurs est que les *kernels* sont limités par la bande passante de mémoire des GPU: l'accès à la mémoire requiert plus de temps que les opérations arithmétiques. Comme l'accès à la mémoire est partagé par les différents fils d'exécution d'un groupe de travail local, le nombre de lectures dans la mémoire globale ne croît pas significativement pour les ordres supérieurs.

En bref, dans la plupart des cas, il est avantageux d'utiliser les opérateurs d'ordres supérieurs, car ceux-ci permettent de réduire la taille de la grille. Par exemple, utiliser un opérateur d'ordre 6 au lieu d'un opérateur d'ordre 2 permet de réduire le nombre de points par longueur d'onde de 22 à approximativement 3. En 3D, une grille comportant 400 fois moins d'éléments peut donc être utilisée. Cependant, les opérateurs d'ordre élevé ne peuvent pas être utilisés dans toutes les circonstances. Par exemple, en présence d'une topographie, d'une surface libre ou en présence de fortes discontinuités dans un matériau, les opérateurs d'ordre élevé introduisent des erreurs (Bohlen & Saenger, 2006; Kristek *et al.*, 2002; Robertsson, 1996). Ainsi, le choix de l'ordre de l'opérateur doit être fait au cas par cas.

4.7.4 Scalabilité sur des grappes de calcul hétérogènes

Le dernier test vise à évaluer la scalabilité de la décomposition de domaine sur de larges grappes de calcul. À cet effet, un test de scalabilité forte a été effectué. La scalabilité forte est la mesure de la variation du temps de calcul pour un modèle de dimension fixe et un nombre croissant de processeurs. Les résultats présentés à la Figure 4.12 ont été obtenus pour un modèle d'une taille de 96x96x1000. Deux différentes grappes de calcul ont été utilisées: Helios de l'Université Laval et Guillimin de l'Université McGill, Canada. Les noeuds d'Helios contiennent 8 GPUs Nvidia K80, qui sont des GPUs doubles, pour un total de 16 GPUs par noeud. Cette grappe de calcul a été utilisée pour tester la scalabilité sur un même noeud, ce qui ne requiert aucune communication MPI (voir

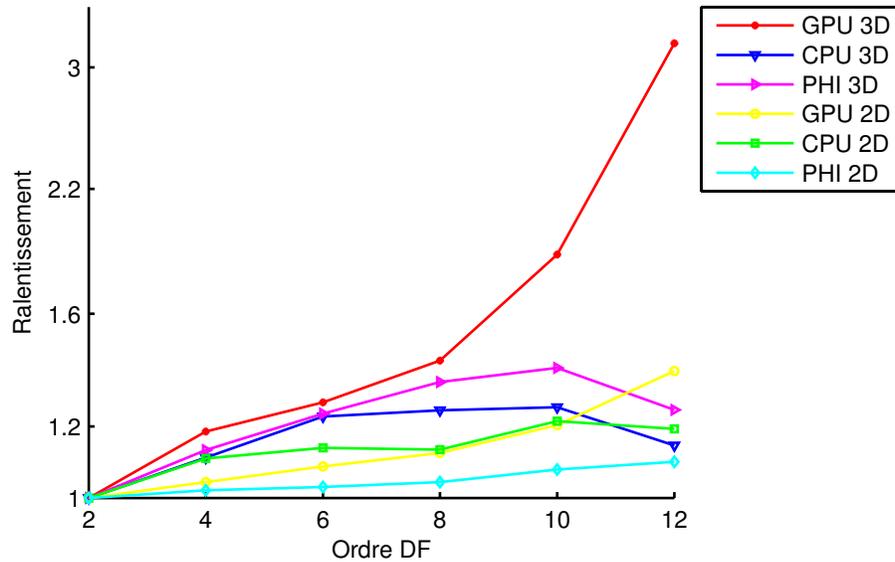


Figure 4.11 – Ralentiement par rapport à un schéma en différences finies d’ordre 2 pour les schémas d’ordre supérieurs.

Figure 4.5). Les noeuds utilisés sur Guillimin contiennent deux types d’engins de calcul: deux CPUs Intel Xeon X5650, avec chacun 6 coeurs cadencés à 2.66 GHz et 12 Mo de cache, ainsi que 2 GPUs Nvidia K20. Cette grappe permet de tester la scalabilité sur plusieurs noeuds, ce qui implique des communications avec MPI.

Les résultats sont présentés à la Figure 4.12. La meilleure scalabilité est obtenue avec les noeuds de Guillimin utilisant les GPUs (étoiles bleues à la Figure 4.12). Celle-ci est presque linéaire sur la plage de noeuds testée. Suprenamment, la scalabilité est moins bonne pour plusieurs GPUs sur le même noeud (noeuds Helios, triangles rouges à la Figure 4.12). Une explication plausible est qu’étant donné la pression croissante sur le CPU lorsque le nombre de GPUs croît, le CPU n’arrive plus à fournir les instructions assez rapidement aux GPUs. Les noeuds de Guillimin ont aussi une meilleure scalabilité lorsque les GPUs sont utilisés comme engin de calcul au lieu du CPU. La scalabilité avec les CPUs demeure tout de même très bonne, au-delà le $N^{4/5}$. Ces résultats sont meilleurs que ceux publiés par Okamoto *et al.* (2011), Rubio *et al.* (2014) et Weiss & Shragge (2013). Cette meilleure performance peut être expliquée par la stratégie de communications et de calculs simultanés décrite à la section 4.5.2.

Ce test de scalabilité montre que le code développé lors de cette thèse est en mesure de paralléliser efficacement le calcul du gradient sur plusieurs noeuds et engins d’une grappe calcul hétérogène. Ceci est requis pour des modèles de grande taille ne pouvant être contenus dans la mémoire d’un

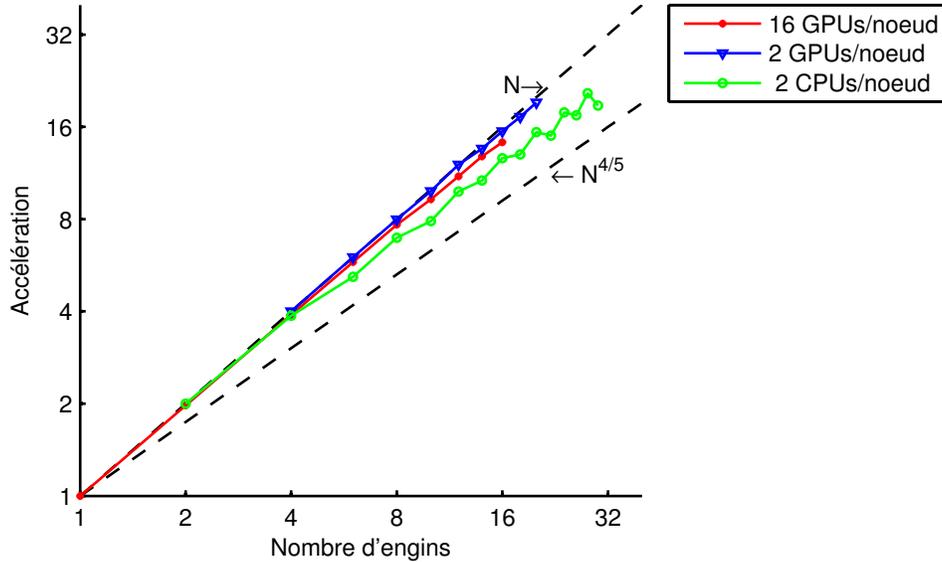


Figure 4.12 – Test de scalabilité forte pour un modèle de 96x96x1000.

seul engin de calcul. Toutefois, la parallélisation des sources devrait être favorisée, et un nombre minimal de noeuds et d'engins de calcul devraient être utilisés lors de la décomposition en domaines. En effet, sans la décomposition de domaine, une simple connexion à faible débit suffit pour utiliser un grand nombre d'ordinateurs. De plus, comme le montre l'accélération en fonction de la taille du modèle à la Figure 4.9, la performance de SeisCL est meilleure lorsque les modèles atteignent une certaine taille. En bref, l'utilisation des deux niveaux de parallélisme permet une certaine flexibilité en ce qui concerne le type de grappe de calcul pouvant être utilisé.

4.8 Conclusion

Une implémentation numérique efficace du calcul de la solution de l'équation d'onde et du gradient de la fonction coût par la méthode adjointe est essentielle pour pouvoir appliquer l'inversion de formes d'ondes complètes à des cas réalistes. L'approche combinée MPI et OpenCL appliquée aux différences finies a prouvé être une approche performante pour ce genre de calcul. Elle permet en particulier de profiter des différents niveaux de parallélisme des plateformes de calcul modernes: du partage des calculs sur différents noeuds d'un super-ordinateur au partage sur différents coeurs d'un processeur de multiple architecture. Le parallélisme inhérent des différences finies permet un gain impressionnant avec une telle stratégie. Notamment, en synchronisant calculs et communications,

une scalabilité linéaire est obtenue pour le partage des mises à jour des variables sismiques sur plusieurs engins de calcul. De plus, l'utilisation de GPUs permet une réduction du temps de calcul substantiel à faible coût. Ceci ouvre la porte à de nouvelles applications de l'inversion de formes d'ondes complètes.

Chapitre 5

Étude de cas: Suivi de la séquestration géologique du CO₂

Suite au développement de la théorie de l'inversion viscoélastique du Chapitre 3 et à l'implémentation numérique présentée au Chapitre 4, la question de la faisabilité de l'inversion multiparamètres peut être abordée. En particulier, les vitesses sismiques et les facteurs d'atténuation peuvent-ils être retrouvés à partir du même jeu de donnée? Ce chapitre aborde ces questions par le biais d'une étude de cas synthétique, dans laquelle l'inversion de formes d'ondes complètes viscoélastique est appliquée au suivi de l'injection de CO₂ par des levés sismiques entre forages.

Ce chapitre est divisé comme suit. Premièrement, le contexte et la problématique de l'étude de cas sont présentés. Puis, une stratégie d'inversion multiparamètres est établie en étudiant le couplage entre les gradients des différentes classes de paramètre. Enfin, les résultats de l'inversion sont présentés et discutés. Les résultats présentés dans ce chapitre font l'objet d'une publication acceptée avec révision modérée par *Geophysical Journal International*:

- Fabien-Ouellet, G., Gloaguen, E., & Giroux, B. (2016). Time domain viscoelastic full waveform inversion *Geophysical Journal International*.

5.1 Problématique

De nombreux domaines bénéficieraient d'une meilleure estimation des propriétés élastiques et viscoélastiques d'un milieu solide, de la géotechnique à la modélisation de réservoirs en passant par la sismologie crustale. Le suivi de la séquestration géologique du CO₂ en est un bon exemple.

En effet, plusieurs organismes internationaux tels que l'Agence internationale de l'énergie (AIE) et le Groupe intergouvernemental d'experts sur l'évolution du climat (GIEC) ont confirmé que le réchauffement climatique est indiscutable et trouve son origine dans l'augmentation de la concentration de gaz à effet de serre rejeté par l'activité humaine dans l'atmosphère (Perozzi *et al.*, 2014). Le GIEC estime que pour limiter cette hausse à 2 °C, objectif adopté au niveau international pour prévenir les répercussions les plus graves du changement climatique, les émissions humaines doivent être limitées à 1000 Gt CO₂ à compter de 2014. Selon les deux organismes, environ 14 % des émissions seront réduites grâce à l'emploi de la technologie du captage et stockage du CO₂. Pour que le stockage géologique du CO₂ ait un impact positif sur l'environnement, le CO₂ doit être stocké dans le sous-sol aussi longtemps qu'il le faut pour que les émissions anthropologiques chutent à des niveaux acceptables.

Pour atteindre cette exigence, il est primordial de s'assurer que le CO₂ reste en place et ne puisse migrer sur de grandes distances, ni verticalement, ni horizontalement. Or, les projets de séquestration ne peuvent avoir assez de puits permettant de faire le suivi temporel du panache de CO₂, en raison des coûts exorbitant de forage. Il est cependant connu que l'injection de CO₂ dans les réservoirs géologiques (que ce soit les anciens réservoirs d'hydrocarbure ou les aquifères salins profonds) cause des changements dans les propriétés viscoélastiques des réservoirs, et donc, des vitesses des ondes P et S (Perozzi *et al.*, 2014; Njiekak *et al.*, 2013) et de leur facteur de qualité (Carcione *et al.*, 2006; Lei & Xue, 2009; Müller *et al.*, 2010). De nombreuses études montrent que la tomographie des premières arrivées permet de suivre l'évolution du panache de CO₂ lors de son injection dans une formation rocheuse (Spetzler *et al.*, 2008; Onishi *et al.*, 2009; Daley *et al.*, 2011; Zhang *et al.*, 2012; Ajo-Franklin *et al.*, 2013). L'inversion de formes d'ondes complètes viscoélastique permettrait une meilleure résolution que la tomographie par tracé de rais, tout en obtenant le changement d'une propriété physique supplémentaire, les facteurs d'atténuation. La présente étude de cas s'inscrit dans ce contexte.

Quoique l'expérience demeure synthétique, le contexte géologique est inspiré d'une étude sur le potentiel de séquestration géologique dans la province de Québec, Canada (Malo & Bédard, 2012). Ainsi, nous tenterons le plus possible de respecter les informations géologiques et pétrophysiques de la région d'études, afin d'étudier un scénario le plus réaliste possible. Le but de ce chapitre demeure néanmoins de vérifier l'applicabilité de la méthode développée aux chapitres précédents, et porte à un plus faible degré sur la question de la séquestration géologique du CO₂.

5.1.1 Contexte géologique

Le site d'études est situé dans le bassin sédimentaire du Cambrien-Ordovicien de la plateforme du Saint-Laurent dans le sud du Québec, Canada. Cette région a été identifiée comme le bassin le plus prometteur pour le stockage géologique du CO₂ dans la province (Malo & Bédard, 2012). Le réservoir potentiel est le groupe du Potsdam, qui repose sur le socle métamorphique Grenvillien du Précambrien. Ce groupe est composé de la formation du Covey Hill (conglomérat quartzeux du Cambrien) et du Cairnside (grès quartzeux de l'Ordovicien). Le groupe du Potsdam est superposé par une succession de formations débutant par le groupe de Beekmantown (grès dolomitique et dolomie) jusqu'aux shales de l'Utica et de Lorraine. L'épaisseur (>800 m) et la perméabilité ($1 \times 10^{-19} \text{ m}^2$) de ces unités dans la région de Bécancour seraient en mesure de prévenir la migration du CO₂ vers la surface, étant donné qu'ils ont maintenu des conditions de surpression dans l'aquifère salin (Ngoc *et al.*, 2014). La Figure 5.1 montre les unités ciblées par l'injection du CO₂, avec une diagraphie des ondes P dans un forage typique de la région.

5.1.2 Construction du modèle synthétique

Le modèle synthétique utilisé dans ce travail a été développé dans une étude préalable réalisée par Perozzi *et al.* (2016). La construction de ce modèle est basée sur une géométrie 2D idéalisée de la séquence sédimentaire montrée à la Figure 5.1. Ce modèle géologique contient quatre couches horizontales: la formation de Theresa, la formation du Cairnside, la formation du Covey Hill et le socle grenvillien. Pour chacune des couches, la moyenne et l'écart-type des propriétés physiques (V_p , V_s , porosité (ϕ) et densité ρ) sont obtenues des diagraphies.

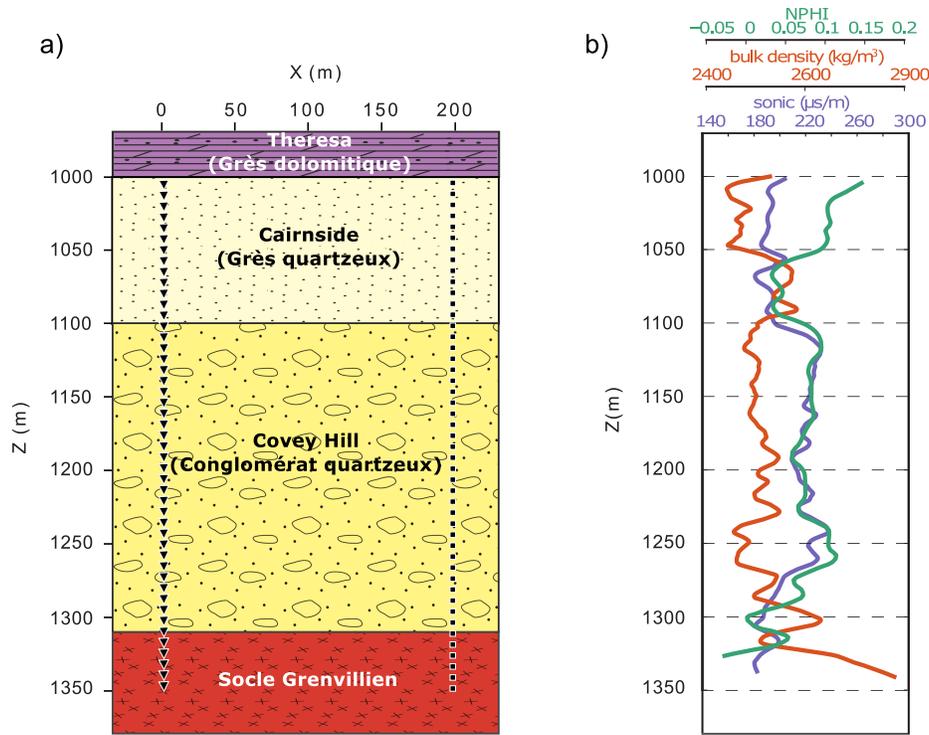


Figure 5.1 – Le modèle géologique montrant le réservoir ciblé pour le CO₂ en a) et une diagraphie sismique représentative de la région en b). Les sources sismiques et les receveurs sont respectivement représentés par des triangles et des carrés noirs.

À partir de ce modèle tabulaire, un modèle de V_p , V_s , τ_p , τ_s et ρ comportant un niveau d'hétérogénéité réaliste a été produit par co-simulation séquentielle gaussienne (Deutsch & Journel, 1992). Le modèle qui en découle est présenté à la Figure 5.2, pour les paramètres d'inversion V_p , V_s , τ_p , et τ_s . Pour ces paramètres, trois zones sont identifiables: une zone de vitesses élevées et d'atténuation moyenne entre 1000 et 1100 m, correspondant à la formation du Cairnside, une zone de basse vitesse et de forte atténuation entre 1100 et 1300 m, correspondant à la formation du Covey Hill, et une zone de grandes vitesses et de faible atténuation sous 1300 m correspondant au socle. Une hétérogénéité substantielle est introduite par la co-simulation dans les trois unités, avec un niveau d'hétérogénéité le plus fort dans la formulation du Covey Hill et le plus faible dans socle grenvillien. Ce modèle sera dénommé le modèle avant injection dans le reste du texte.

5.1.3 Modélisation du transport de CO₂

Le transport du panache de CO₂ est modélisé par le solveur à équilibre vertical de la boîte à outils Matlab *Reservoir Simulation Toolbox* (Lie *et al.*, 2012). Les paramètres du modèle hétérogène

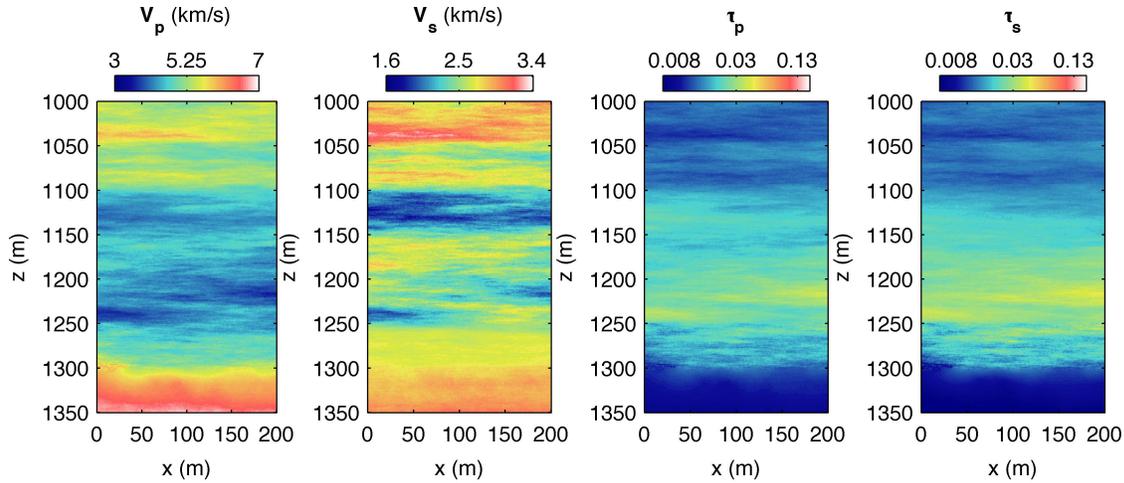


Figure 5.2 – Modèle avant injection des 4 paramètres sismiques que l’on veut imager par inversion.

obtenus par co-simulation sont utilisés comme référence. Cette modélisation est tirée de Perozzi *et al.* (2014). L’injection de CO₂ dans les formations du Postdam est simulée pour un temps d’injection de 15 ans, à un taux d’injection de 45 t/jour, un taux comparable à l’injection moyenne au site de Ketzin (Martens *et al.*, 2012). Le puits d’injection est situé à la gauche du modèle. La concentration en CO₂ qui en résulte est présentée à la Figure 5.3. Le panache après 15 ans atteint le puits d’écoute (droite du modèle). La saturation maximale en CO₂ dépasse les 40 %.

L’influence du CO₂ sur les propriétés sismiques des formations rocheuses est modélisée par la méthode de Carcione *et al.* (2006) basée sur le modèle de saturation partielle de White (1975). Les valeurs de taille de zone saturée (*patch size*) et la fréquence centrale sont choisies arbitrairement, soit respectivement 5 mm et 800 Hz. Ces valeurs ne sont pas calibrées et sont utilisées pour produire des changements de vitesse et de facteur de qualité substantiels, dans le but de tester la méthodologie. Comme le fluide des pores ne devrait pas avoir d’impact sur les ondes de cisaillement, aucun changement de V_s et Q_s n’est modélisé. Dans un scénario réel, d’autres facteurs, tel le changement de pression des pores, affectent la vitesse des ondes S et devraient être pris en compte. Le modèle après injection est présenté à la Figure 5.3. La forme du panache de CO₂ est clairement identifiable sur l’image de Q_p , son pourtour étant manifeste entre 1150 et 1250 m de profondeur. La présence de CO₂ conduit à une réduction des vitesses et du facteur de qualité, jusqu’à 24% et 73 % respectivement. Quoique ces changements semblent assez élevés, ces plages de valeur sont représentatives de ce qui peut être observé sur le terrain (communication personnelle avec Benjamin Ajo-Franklin). Ceci nous permet aussi d’étudier l’impact de différentes magnitudes d’atténuation sur l’inversion de formes d’ondes complètes viscoélastique, ce qui demeure le but premier de ce chapitre.

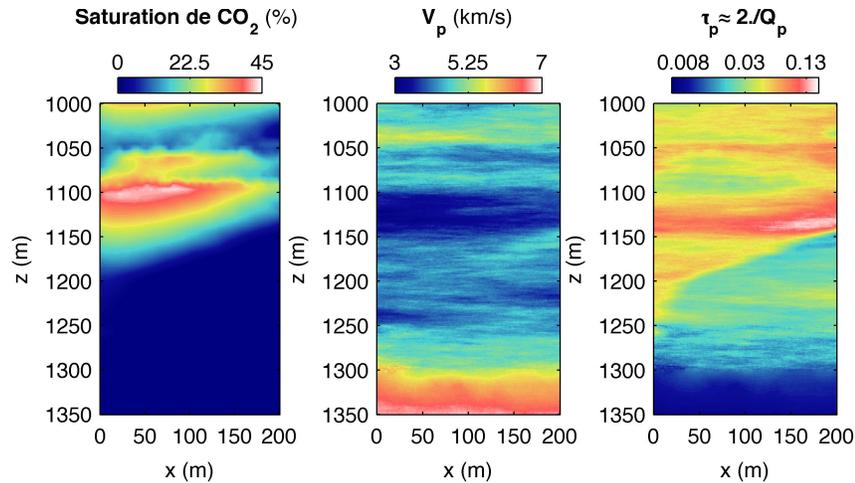


Figure 5.3 – Les saturations en CO_2 après 15 ans d’injection (gauche), les vitesses (centre) et le niveau d’atténuation (droite) des ondes P suite à l’injection.

5.1.4 Suivi sismique entre forages

Le suivi sismique du panache de CO_2 est réalisé par des levés sismiques entre forages. Deux levés sont simulés: un avant injection et un après 15 ans d’injection. Ces deux levés partagent la même configuration d’acquisition. Les sources sismiques sont positionnées à tous les mètres le long du forage d’injection, situé à $x=0$ m à la Figure 5.1. Les positions de tir débutent au toit de la formation du Cairnside à la profondeur 1000 m, et vont jusqu’au socle rocheux à une profondeur de 350 m. Deux types de sources sont utilisés: une force en z et une source explosive, toutes deux modélisées par une impulsion de Ricker avec une fréquence centrale de 500 Hz. Ces sources sont représentatives des sources utilisées pour les levés entre forages (Daley & Cox, 2001; Daley *et al.*, 2007; Ajo-Franklin *et al.*, 2013). Les receveurs sont situés dans le forage à $x=200$ m. Des géophones deux composantes qui mesurent la vitesse des particules en x et z (v_x et v_z) sont placés au même intervalle que les sources (1 m).

La propagation des ondes pour les modèles avant et après injection est modélisée avec le code de simulation décrit au chapitre précédent afin de constituer le jeu de données synthétiques \mathbf{d} , soit les vitesses des particules qui seraient mesurées sur le terrain pour les modèles vrais. Un opérateur en différences finies d’ordre 8 est utilisé, soit l’ordre le plus efficace en terme de temps de calcul avec SeisCL. Avec une vitesse minimale des ondes S de 1690 m/s, la taille du pas spatial de la grille est choisie à 0.5 m pour minimiser la dispersion numérique jusqu’à 1000 Hz (équation 2.42). Afin de respecter le critère de stabilité pour une vitesse maximale de 7300 m/s, le pas de temps temporel

est de 35 μs (équation 2.43). Un seul mécanisme d'atténuation est utilisé dans cet exemple, avec une fréquence de $2\pi/\tau_{\sigma l} = 500$ Hz. À noter que ce choix n'est basé sur aucune information terrain et devrait être révisé dans le cas d'une application réelle. Les données de référence ainsi que les inversions subséquentes sont toutes modélisées avec les mêmes paramètres.

5.2 Développement d'une stratégie d'inversion multiparamètres

Comme le montrent plusieurs études sur le sujet (voir en particulier Kamei & Pratt (2013)), l'interférence entre les parties réelles et imaginaires des vitesses sismiques peut être problématique en introduisant d'importantes anomalies lors de l'inversion du facteur de qualité Q . En fait, l'interdépendance des paramètres est un caractère général de l'inversion de formes d'ondes complètes multiparamètres, c'est-à-dire lorsque plusieurs classes de paramètres sont inversées simultanément. Operto *et al.* (2013) listent plusieurs stratégies pour minimiser l'interférence entre les classes de paramètres: le choix d'une paramétrisation appropriée, l'utilisation de la Hessienne, une méthodologie orientée sur les données dans laquelle différentes parties des données sont sélectionnées pour chaque classe de paramètre et une méthodologie orientée sur le modèle dans laquelle les paramètres sont inversés de façon hiérarchique, du plus dominant au moins dominant. Ce qui suit vise à étudier l'interférence entre les classes de paramètres afin d'établir une méthodologie d'inversion à la fois guidée par les données et le modèle, qui permettra de procéder à l'inversion des levés sismiques entre forages.

5.2.1 Interdépendance des paramètres viscoélastique

L'équation d'onde viscoélastique isotrope dépend d'au moins cinq classes de paramètres physiques, par exemple V_p , V_s , Q_p , Q_s et ρ . Comme les niveaux d'atténuation sont fortement reliés à la magnitude du facteur de qualité (pour un mécanisme, $Q \approx 2/\tau$), nous choisissons plutôt comme paramétrisation d'inversion (V_p , V_s , τ_p , τ_s et ρ). La forme du profil en fréquence, contrôlée par le nombre de mécanismes de Maxwell et leur temps de relaxation $\tau_{\sigma l}$, est considérée fixe et connue dans le restant du travail. Nous faisons de même pour la densité, qui est difficilement reconstruite par l'inversion de formes d'ondes complètes. L'influence du choix de la paramétrisation n'est pas étudiée plus en détail, mais devrait faire l'objet de travaux ultérieurs.

Tableau 5.1 – Résumé des paires de paramètres pour lesquels le couplage est étudié.

| Figure | Type de source | Paramètre 1 | Paramètre 2 |
|--------|----------------|---------------------------|-------------------------------|
| 5.4 | explosive | $\Delta V_p = 1.4\%$ | $\Delta \tau_p = (50, 300)\%$ |
| 5.5 | force en z | $\Delta V_s = 2.5\%$ | $\Delta \tau_s = (50, 300)\%$ |
| 5.6 | force en z | $\Delta V_p = (1.4, 8)\%$ | $\Delta V_s = 2.5\%$ |
| 5.7 | explosive | $\Delta V_p = 1.4\%$ | $\Delta V_s = (2.5, 5)\%$ |

Le couplage entre les quatre paramètres choisis (V_p , V_s , τ_p , τ_s) est ici étudié dans le contexte des levés entre forages, soit la configuration utilisée pour la présente étude synthétique. Pour cette expérience, une impulsion de Ricker avec une fréquence centrale de 25 Hz est utilisée, contrairement à l'étude de cas. Les sources sont positionnées à tous les 20 m le long du côté gauche des modèles et les géophones sont positionnés au même intervalle le long du côté droit. De façon similaire aux levés de suivi sismiques, deux types de sources sont utilisés: une source explosive qui émet principalement des ondes P et une force en z qui émet principalement des ondes S .

Pour étudier le couplage entre les paramètres, ceux-ci sont analysés deux à deux. Pour chaque couple de paramètres, une perturbation est appliquée aux modèles homogènes des deux paramètres en question à des positions différentes. Les paramètres du modèle homogène sont $V_p = 3500 \text{ m/s}$, $V_s = 2000 \text{ m/s}$, $\rho = 2000 \text{ kg/m}^3$, $\tau_p = 0.02$, et $\tau_s = 0.02$. Le gradient du modèle homogène par rapport au modèle perturbé est ensuite calculé. Pour deux paramètres au couplage nul, leur gradient devrait atteindre une valeur maximale autour des perturbations respectives. Au contraire, en présence d'interférence, le gradient sera perturbé et ne pointera pas dans la direction de la perturbation. Dans ce qui suit, trois couples de paramètres sont étudiés: (V_p, τ_p) pour une source explosive, (V_s, τ_s) pour une source en z et (V_p, V_s) pour des sources explosives et pour des sources orientées en z . Le Tableau 5.1 résume les paramètres de ces expériences.

Le couplage entre V_p et τ_p est illustré à la Figure 5.4. La source utilisée est une source explosive. Deux cas sont considérés: une faible et une forte perturbation de τ_p , avec $\Delta \tau_p = 0.01$ et 0.06 (50 et 300 %), respectivement. Dans les deux cas, la perturbation ΔV_p reste constante à 50 m/s (1.4 %). Les gradients sont présentés à la Figure 5.4. Pour la faible perturbation de τ_p , le gradient de τ_p est complètement dominé par la perturbation ΔV_p , la plus forte composante du gradient étant localisée autour de la perturbation de V_p . Au contraire, le gradient de V_p contient une composante nulle autour de la perturbation de τ_p dans ce cas. Pour la forte perturbation $\Delta \tau_p$, l'effet du couplage sur le gradient de τ_p diminue, tout en demeurant visible. Dans ce cas, le gradient de V_p commence à

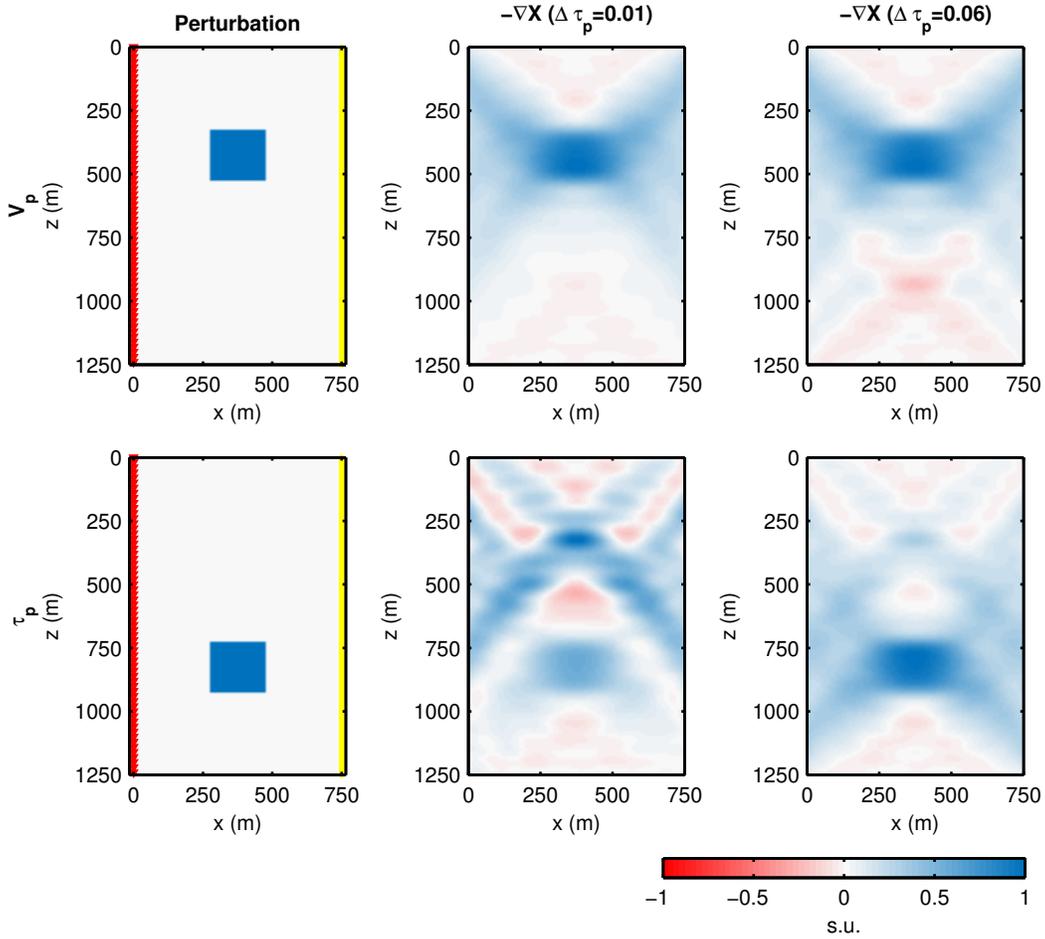


Figure 5.4 – Effet du couplage entre V_p (rangée du haut) et τ_p (rangée du bas) pour une source explosive. Position des perturbations (colonne de gauche) et gradients normalisés pour un faible $\Delta \tau_p$ de 0.01 (50 %) (colonne du centre) et un fort $\Delta \tau_p$ de 0.06 (300 %). Les sources sont en rouge et les receveurs sont en jaune.

présenter une composante dans la direction de l’anomalie de τ_p , qui reste toutefois faible comparée à la composante localisée autour de la perturbation ΔV_p . On peut conclure que la vitesse des ondes P est ici le paramètre dominant par rapport à τ_p . Cet exemple montre aussi qu’une faible erreur du modèle de vitesse V_p peut significativement perturber le gradient du niveau d’atténuation, ce qui pourrait limiter la convergence de l’inversion pour τ_p .

Les résultats pour la même expérience entre τ_s et V_s sont présentés à la Figure 5.5. Dans ce cas, une source orientée en z est utilisée. La perturbation ΔV_s est gardée constante à 50 m/s (2.5%) et deux magnitudes sont utilisées pour $\Delta \tau_s$: 0.02 et 0.06 (50 and 300 %). Un comportement similaire est observé: V_s est peu affecté par le couplage pour les deux tailles de perturbation, tandis que τ_s est fortement affecté par la perturbation ΔV_s . Pour la perturbation faible, la composante du gradient

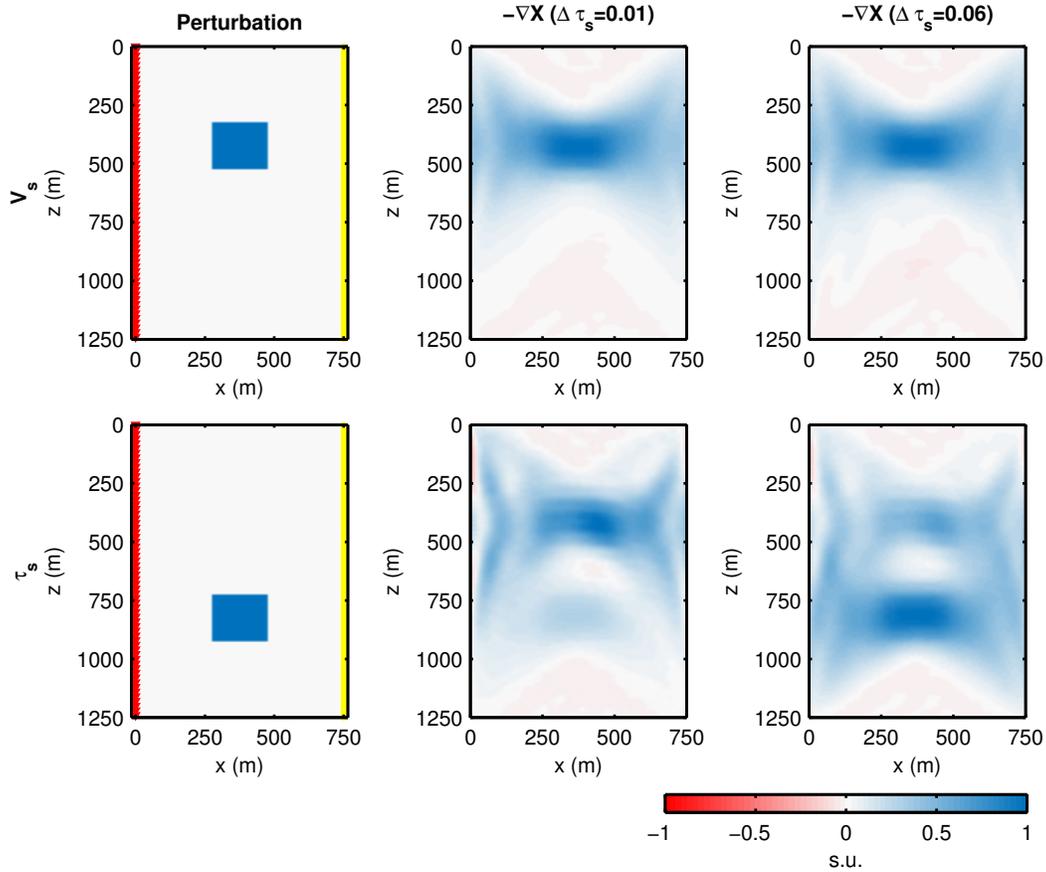


Figure 5.5 – Effet du couplage entre V_s (rangée du haut) et τ_s (rangée du bas) pour une source orientée en z . Position des perturbations (colonne de gauche) et gradients normalisés pour un faible $\Delta\tau_s$ de 0.01 (50 %) (colonne du centre) et un fort $\Delta\tau_s$ de 0.06 (300 %). Les sources sont en rouge et les receveurs sont en jaune.

de τ_s est plus forte autour de la perturbation en V_s . Pour la perturbation plus forte, la correction est plus forte cette fois-ci autour de $\Delta\tau_s$, même si une composante non négligeable du gradient est encore localisée autour de la perturbation en V_s .

Un couplage existe aussi entre V_p et V_s . Pour ces deux paramètres, deux expériences sont effectuées: une pour des sources orientées en z et une pour des sources explosives. Les résultats sont présentés respectivement aux Figures 5.6 et 5.7. Pour la source orientée en z , ΔV_s reste fixe à 50 m/s (2.5 %) avec une faible et forte perturbation ΔV_p de 50 et 200 m/s (1.4 et 8%) respectivement. Pour ce type de source, V_s est le paramètre dominant: son gradient n'est affecté par aucune des deux tailles de perturbation ΔV_p . Au contraire, le gradient de V_p est affecté par la perturbation de V_s , avec un effet moindre pour la perturbation plus forte. À noter que l'interférence dans ce cas n'est pas aussi cohérente que l'interférence entre les vitesses et l'atténuation. Pour la source

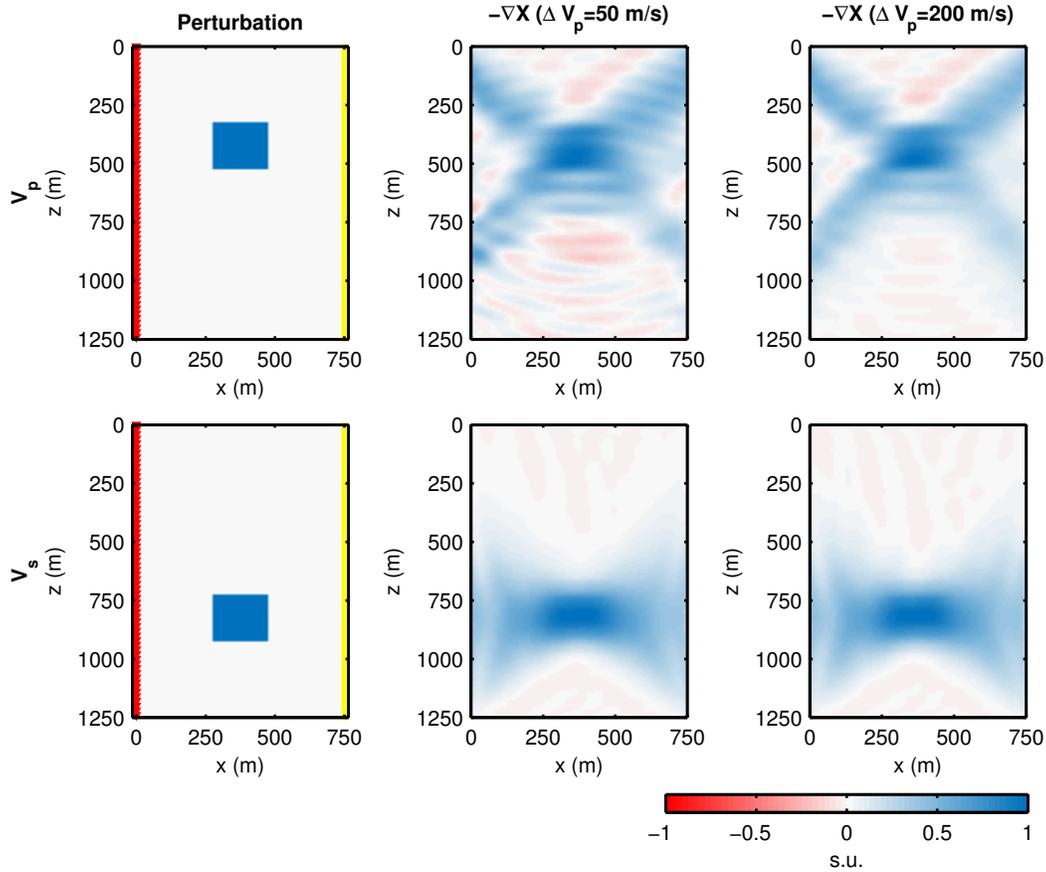


Figure 5.6 – Effet du couplage entre V_p (rangée du haut) et V_s (rangée du bas) pour une source orientée en z . Position des perturbations (colonne de gauche) et gradients normalisés pour un faible ΔV_p de 50 m/s (1.4%) (colonne du centre) et un fort ΔV_p de 200 m/s (8 %). Les sources sont en rouge et les receveurs sont en jaune.

explosive, ΔV_p est fixé à 50 m/s (1.4 %) et deux tailles de perturbation sont considérées pour ΔV_s : 50 m/s et 100 m/s (2.5 et 5%). Les résultats sont présentés à la Figure 5.7. Ceux-ci indiquent que V_p est le paramètre dominant pour une source explosive, quoique son gradient est affecté par la plus grande des perturbations de V_s . Le gradient de V_s est quant à lui incohérent pour les deux tailles de perturbations. Ceci peut être compris grâce au diagramme de rayonnement d'une source explosive; celle-ci ne produit que des ondes P (voir Heelan (1953) et Gibson & Richard (1994) sur le sujet). Les ondes S ne sont présentes que par la conversion de mode. En raison de l'absence des ondes S , le gradient ne peut être cohérent pour V_s . Pour la source orientée en z , les ondes S sont dominantes pour les petits angles tandis que les ondes P deviennent dominantes pour des angles supérieurs à $\pm 45^\circ$. Ainsi, dans ce cas, les deux types d'ondes sont produits par la source et le gradient de V_p comme le gradient de V_s sont cohérents et pointent dans la bonne direction.

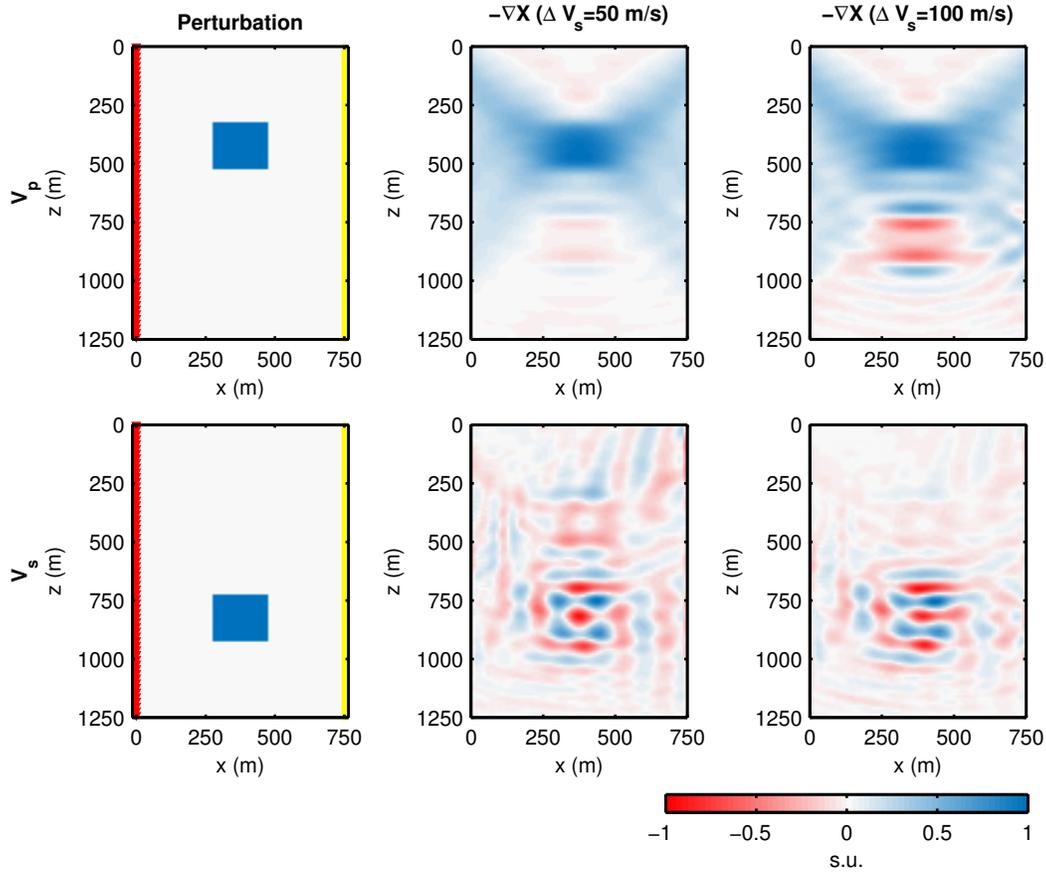


Figure 5.7 – Effet du couplage entre V_p (rangée du haut) et V_s (rangée du bas) pour une source explosive. Position des perturbations (colonne de gauche) et gradients normalisés pour un faible ΔV_s de 50 m/s (2.5%) (colonne du centre) et un fort ΔV_s de 100 m/s (5 %). Les sources sont en rouge et les receveurs sont en jaune.

Le couplage entre d'autres pairs de paramètre n'est pas présenté ici. Or, ils ont le même comportement. Pour la source explosive, la hiérarchie de sensibilité des paramètres est $V_p \rightarrow V_s \rightarrow \tau_p \rightarrow \tau_s$, du plus sensible, au moins sensible. Pour la source orientée en z , l'ordre d'importance est plutôt $V_s \rightarrow V_p \rightarrow \tau_s \rightarrow \tau_p$. Ces résultats suggèrent deux stratégies pour limiter l'effet du couplage entre les paramètres lors de l'inversion. La première est orientée sur les données: l'inversion de V_p et τ_p est effectuée avec le jeu de données de la source explosive, tandis que l'inversion de V_s et τ_s est effectuée à l'aide du jeu de données des sources orientées en z . La seconde stratégie est hiérarchique: les paramètres dominants (V_p et V_s) sont inversés avant les paramètres moins sensibles (τ_p et τ_s).

5.2.2 Stratégie d'inversion

Suite à cette expérience, l'approche d'inversion multiparamètres suivante est adoptée pour l'étude de cas. L'inversion est basée sur l'algorithme de descente par gradient stochastique décrit à la section 2.3.4. Nous utilisons un sous-ensemble de sources tiré aléatoirement selon une distribution uniforme. Ce sous-ensemble change à chaque itération. Nous utilisons une variante du SGD avec une grandeur de sous-ensemble croissante, comme décrite par Friedlander & Schmidt (2012) et appliquée à l'inversion de formes d'ondes complètes par van Leeuwen & Herrmann (2013). L'augmentation graduelle de la taille du sous-ensemble empêche le plafonnement de la convergence lors de l'inversion. Le calcul du gradient étant moins coûteux plus le nombre de sources est faible, l'inversion par gradient stochastique est beaucoup plus rapide que l'algorithme de la plus forte descente traditionnel. De plus, l'acquisition des données sismiques est conçue pour être hautement redondante. Ainsi, le sous-échantillonnage des sources n'affecte pas trop la convergence, puisqu'il ne diminue pas significativement la quantité d'information du jeu de données.

Afin d'éviter le saut de cycle, l'approche hiérarchique multi-échelles de Bunks *et al.* (1995) est utilisée: l'inversion débute par les basses fréquences, et la fréquence inversée augmente au fur et à mesure des itérations. Le gradient est calculé dans le domaine des fréquences par l'approche en fréquence décrite à la section 4.5.4, Algorithme 9. Étant donné les longueurs d'onde différentes des ondes S et des ondes P , deux bandes de fréquences différentes sont utilisées pour leurs paramètres respectifs. Pour V_s , des fréquences entre 80 et 400 Hz tous les 20 Hz sont utilisées, tandis que des fréquences entre 150 Hz et 950 Hz tous les 50 Hz sont utilisées pour V_p . Comme l'inversion des niveaux d'atténuation semble bénéficier d'une grande largeur de bande, les deux plages de fréquences précédentes ont été divisées en deux bandes de fréquences: 80 à 200 Hz et 200 à 400 Hz pour τ_s et 150 à 500 Hz et 500 à 950 Hz pour τ_p . Pour chacune de ces bandes de fréquences, toutes les fréquences sont inversées simultanément. Dans les cas des vitesses V_p et V_s , chaque fréquence est inversée séparément. Chacune des étapes d'inversion pour une fréquence ou une bande de fréquences contient 40 itérations de descente par gradient stochastique, avec un sous-ensemble de sources différent. Ce nombre de sources croît avec la fréquence, de 24 sources pour la fréquence la plus basse, à 76 pour la fréquence la plus élevée.

La réduction du couplage des paramètres se fait avec les deux stratégies proposées à la section précédente. L'inversion de V_p et τ_p se fait avec le jeu de données des sources explosives tandis que

l'inversion de V_s et τ_s se fait avec le jeu de données contenant les sources orientées en z . Pour chaque itération de l'inversion, V_s est premièrement inversé, suivit par V_p . Les niveaux d'atténuation τ_s et τ_p sont inversés après l'inversion de toutes les fréquences pour V_s et V_p . De plus, en focalisant sur un type de paramètre à la fois, cette inversion séquentielle permet de contourner le problème du dimensionnement du gradient entre les classes de paramètres (Kamei & Pratt, 2013). La méthodologie d'inversion est résumée à l'Algorithme 12.

Algorithme 12 Méthodologie d'inversion

```

1: Définir  $freqvs = [80 : 20 : 400]$ 
2: Définir  $freqvp = [150 : 50 : 950]$ 
3: Définir  $freqall = freqvs + freqvp$ 
4: for all  $freq \in freqall$  do
5:   if  $freq$  in  $freqvs$  then
6:     SGD( $V_s, freq, 8 + 3.3ns, zdata, 40$ )
7:     Incrémenter  $ns$ 
8:   if  $freq$  in  $freqvp$  then
9:     SGD( $V_p, freq, 8 + 3.3np, pdata, 40$ )
10:    Incrémenter  $np$ 
11: SGD( $\tau_s, freqvs < 200, 120, zdata, 40$ )
12: SGD( $\tau_s, freqvs \geq 200, 120, zdata, 40$ )
13: SGD( $\tau_p, freqvp < 400, 120, pdata, 40$ )
14: SGD( $\tau_p, freqvp \geq 400, 120, pdata, 40$ )

15: procedure SGD( $param, freqs, nsource, data, niter$ )
16:   while  $iter < niter$  do
17:     Choisir  $nsource$  from  $data$ 
18:     Calculer  $gradient$  for  $freqs$  and  $selected data$ 
19:     Recherche linéaire sur  $gradient$ 
20:     Mise à jour du  $model$ 
21:     Incrémenter  $iter$ 

```

5.3 Suivi temporel du panache de CO₂

5.3.1 Inversion du modèle avant injection

Le modèle initial pour l'inversion avant l'injection est une version lissée du vrai modèle (deuxième colonne de la Figure 5.9). Ce modèle pourrait être obtenu de façon réaliste par la tomographie des premières arrivées et des amplitudes basée sur le tracé de rai. Ces modèles contiennent des variations à grande échelle et les différentes couches géologiques sont visibles. Le but de l'inversion de formes

d'ondes complètes est ici d'améliorer la résolution spatiale de chacun des paramètres et de mieux reproduire la distribution des valeurs, principalement des valeurs extrêmes.

Les regroupements à déport nul du modèle vrai avant injection sont présentés à la Figure 5.8, première colonne. La profondeur de la source et des récepteurs sont les mêmes pour chaque trace de ces regroupements. Les enregistrements montrent tous des arrivées sismiques très complexes, comprenant des ondes directes, des ondes réfractées, des ondes réfléchies, des ondes diffractées et des ondes converties. Cette complexité est une conséquence directe de la variabilité spatiale élevée du modèle synthétique: les vitesses peuvent varier de plus de 200 % sur une courte distance. Pour cette raison, ce jeu de données représente un défi pour l'inversion de formes d'ondes complètes. Les premières arrivées des ondes P et des ondes S ont été calculées par tracé de rais. Celles-ci sont indiquées par les lignes bleues et rouges respectivement à la Figure 5.8. Les deux types d'arrivées sont présents sur les regroupements sismiques de la force en z , quoique les ondes S sont beaucoup plus énergétiques. Sur les regroupements des sources explosives, seules les premières arrivées des ondes P sont clairement identifiables, ainsi que des ondes converties de plus faible amplitude. Étant donné que les ondes P dominent l'énergie des regroupements des sources explosives et que les ondes S dominent les regroupements des sources en z , le couplage des paramètres devrait être minimisé en utilisant séparément chaque jeu de données pour l'inversion des ondes P et S .

En comparant les regroupements à déport nul du modèle initial et du modèle vrai (première et deuxième colonnes de la Figure 5.8, respectivement), on constate que les données ne contiennent que les premières arrivées pour le modèle initial tandis qu'elles contiennent des formes d'onde très complexes pour le modèle vrai. Les résidus (troisième colonne) sont fortement atteints par le saut de cycle: la superposition des données du modèle initial et du modèle vrai est clairement identifiable. En bref, le modèle initial ne prédit que très mal les données enregistrées.

Le modèle inversé est montré à la troisième colonne de la Figure 5.9. Tant pour V_p que pour V_s , les détails d'une très fine résolution sont bien reconstruits. Seuls les nombres d'ondes très élevés ne sont pas retrouvés, ce qui est normal étant donné la bande de fréquences limitée de la source sismique. L'inversion a sans contredit convergé à la bonne solution dans les deux cas. Ceci est confirmé par le graphique quantile-quantile de la Figure 5.10. Ce graphique est construit en calculant les percentiles des modèles vrais et inverses, et en assignant le premier à l'axe des abscisses et le second à l'axe des ordonnées. Si les deux modèles partagent la même distribution de valeurs, leurs quantiles devraient

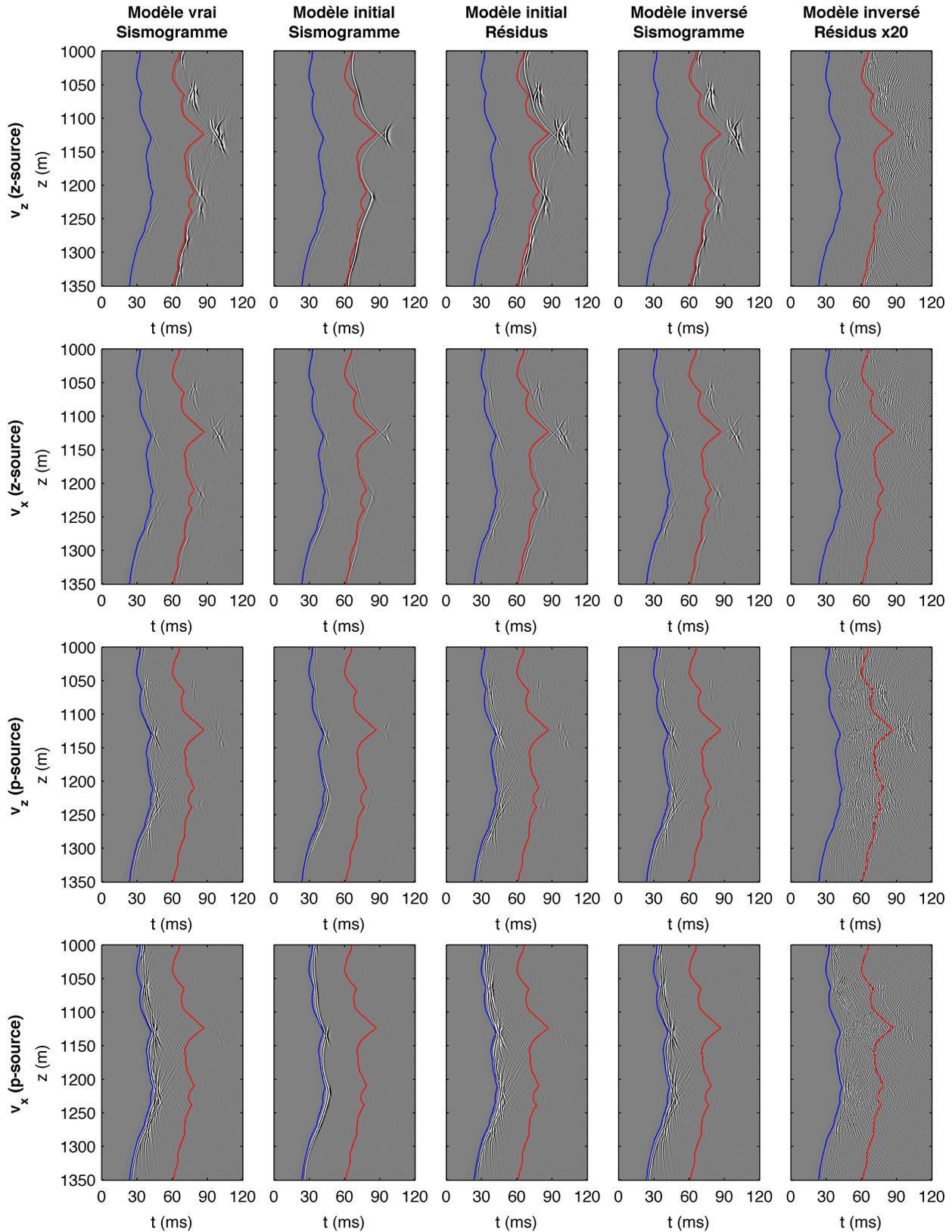


Figure 5.8 – Regroupement à départ nul pour le vrai modèle avant injection (première colonne), le regroupement à départ nul et les résidus du modèle initial (deuxième et troisième colonnes) et le regroupement à départ nul et les résidus du modèle inversé (quatrième et cinquième colonnes). Les première et deuxième rangées présentent les composantes v_z et v_x de la source orientée en z , tandis que les troisième et quatrième rangées présentent les composantes v_z et v_x de la source explosive. Les lignes bleues et rouge indiquent les premières arrivées des ondes P et S , respectivement.

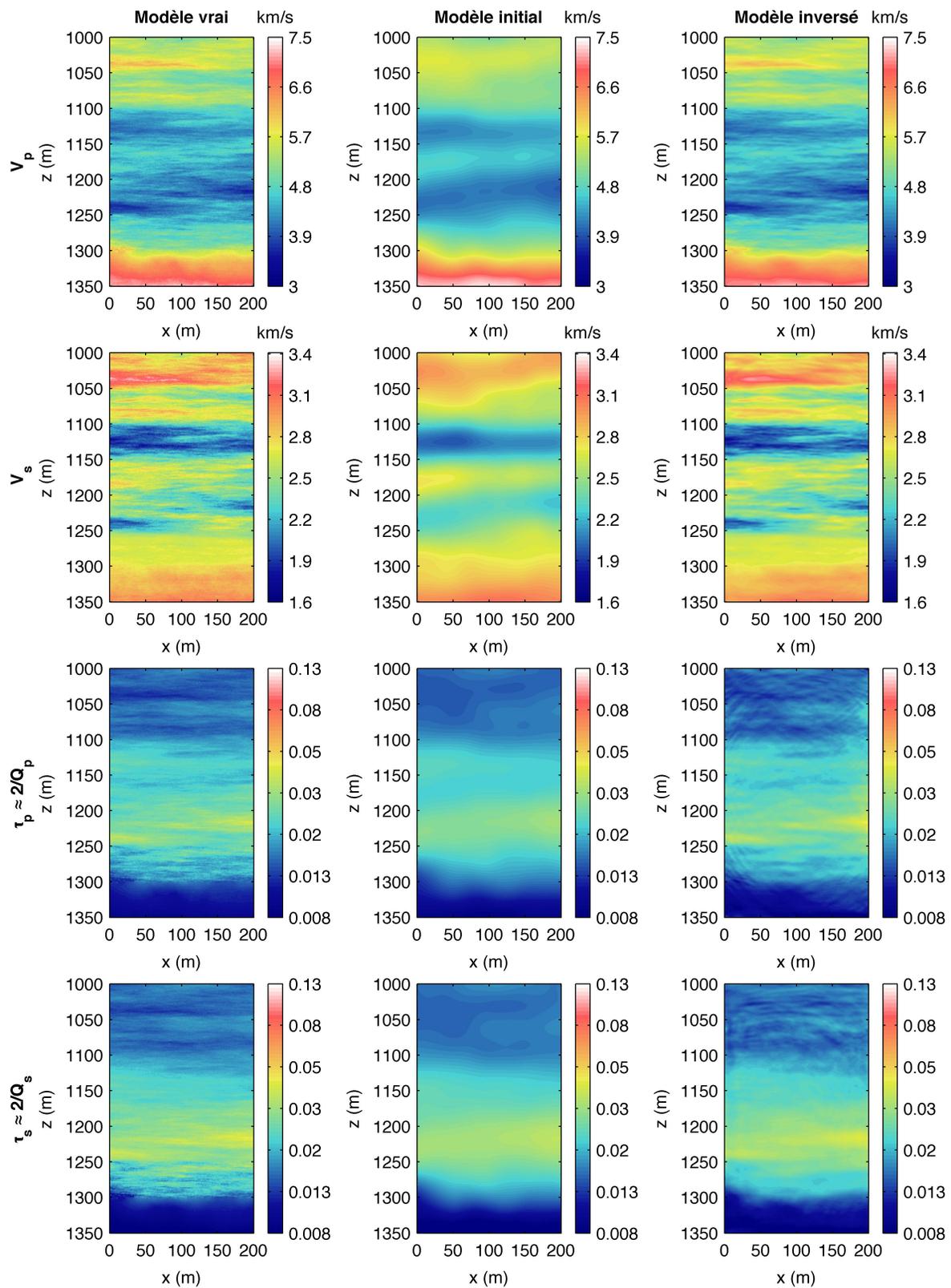


Figure 5.9 – Le modèle vrai avant injection (première colonne), le modèle initial (deuxième colonne) et le modèle inversé (troisième colonne). La vitesse des ondes P est présentée à la première rangée, la vitesse des ondes S à la deuxième rangée et leurs niveaux d'atténuation aux troisième et quatrième rangées.

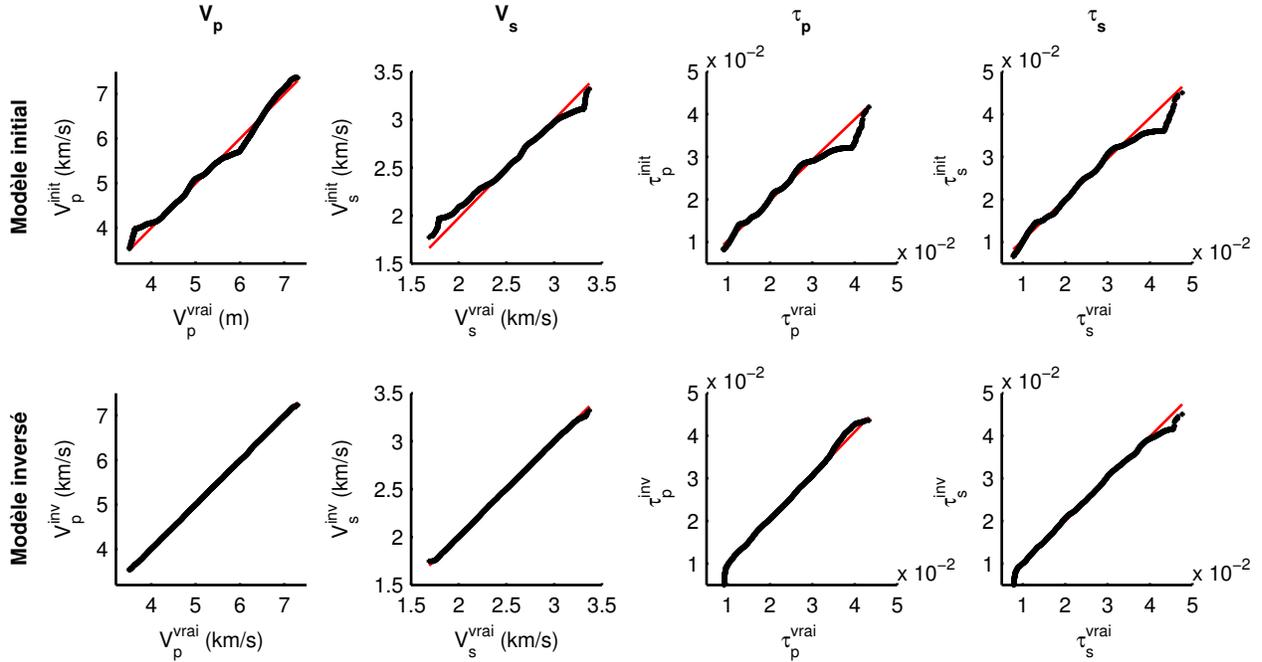


Figure 5.10 – Graphique quantile-quantile entre le modèle vrai avant injection et les modèles initiaux et inversés. Les lignes rouges représentent une correspondance exacte entre les deux distributions et les points noirs représentent les quantiles calculés.

être équivalents. Dans le cas contraire, par exemple, si un modèle est plus lisse que l'autre ou présente un biais, le graphique quantile-quantile déviara d'une droite à 45°. Avant l'inversion, les percentiles extrêmes divergent de façon notable du modèle vrai sur ce graphique. Ceci est logique; le modèle initial est une version lisse du modèle vrai, reproduisant mal les valeurs extrêmes. Après l'inversion, la concordance entre les distributions des paramètres inversés et vrais est presque exacte, ce qui montre que la plus grande partie de la variabilité spatiale des vitesses est bien reconstruite.

D'autre part, τ_p et τ_s ne sont pas reconstruits avec le même niveau de précision que les vitesses. Une partie des détails de haute résolution est reconstruite, en particulier dans les régions de forte atténuation. Cependant, les paramètres d'atténuation sont victimes du couplage avec les petites erreurs toujours contenues dans les modèles inversés de V_p et V_s . Comme il a été illustré à la dernière section, l'interférence se produit même pour de faibles erreurs de vitesse. Ainsi, pour reconstruire les niveaux d'atténuation, des modèles de vitesse très précis sont requis. La conséquence du couplage sur l'inversion de τ_p et τ_s peut être observée à la Figure 5.10 pour les faibles quantiles: l'inversion détériore la correspondance. Cette figure montre aussi que les fortes valeurs ne sont pas bien reconstruites, mais que l'inversion améliore la reconstruction des valeurs intermédiaires. Quoique

le couplage introduit quelques anomalies, l'inversion de formes d'ondes complètes améliore quand même la résolution par rapport au modèle initial de τ_p et τ_s .

En comparant les regroupements sismiques du modèle vrai et inversé (première et troisième colonnes de la Figure 5.8), on constate que le champ d'onde est très bien reproduit et contient toutes les arrivées complexes des données enregistrées. L'amplitude des résidus (cinquième colonne) a fortement diminué suite à l'inversion. De plus, la majorité de l'énergie des résidus correspond à des ondes converties ou dispersées à multiples reprises. Ces types d'arrivées sont généralement très difficilement reproduits par l'inversion de formes d'ondes complètes (Virieux & Operto, 2009). De plus, la majorité de l'énergie résiduelle est de plus haute fréquence que la fréquence dominante des enregistrements. En raison du saut de cycle, ces hautes fréquences sont plus difficilement inversibles. De plus, les fréquences au-dessus de 400 Hz n'ont même pas été inversées pour les ondes S . Par conséquent, la majorité de l'énergie des résidus devrait être de plus haute fréquence, ce qui est observé. L'amélioration de la prédiction des données enregistrées du modèle inversé par rapport au modèle initial mène à la conclusion que l'inversion de formes d'ondes complètes a bel et bien convergé pour le modèle avant injection.

5.3.2 Inversion du modèle après injection

Comme pour l'inversion avant injection, le modèle initial pour l'inversion des données après 15 ans d'injection est une version lisse du modèle vrai pour V_p et τ_p (première et deuxième rangée de la Figure 5.11). Le modèle inversé avant injection pour V_s et τ_s est utilisé dans ce cas, comme il a été supposé que les propriétés des ondes S ne sont pas affectées par la présence du CO_2 . Par conséquent, seuls V_p et τ_p sont inversés après l'injection. Contrairement au modèle avant injection, les regroupements sismiques de la source explosive du modèle initial (deuxième colonne de la Figure 5.12) contiennent non seulement des arrivées directes des ondes P , mais aussi une quantité non négligeable d'ondes converties. Ceci est la conséquence d'utiliser le modèle inversé de V_s , qui reproduit la majorité de l'hétérogénéité spatiale du modèle vrai. Ainsi, même si V_p est le principal paramètre d'intérêt, il n'est pas toujours possible de négliger l'impact de V_s .

Les résultats de l'inversion sont montrés à la troisième colonne de la Figure 5.11. Encore une fois, V_p est presque parfaitement reconstruit et le graphique quantile-quantile (Figure 5.13) indique que les deux distributions de valeurs sont pratiquement identiques. De plus, les détails haute résolution

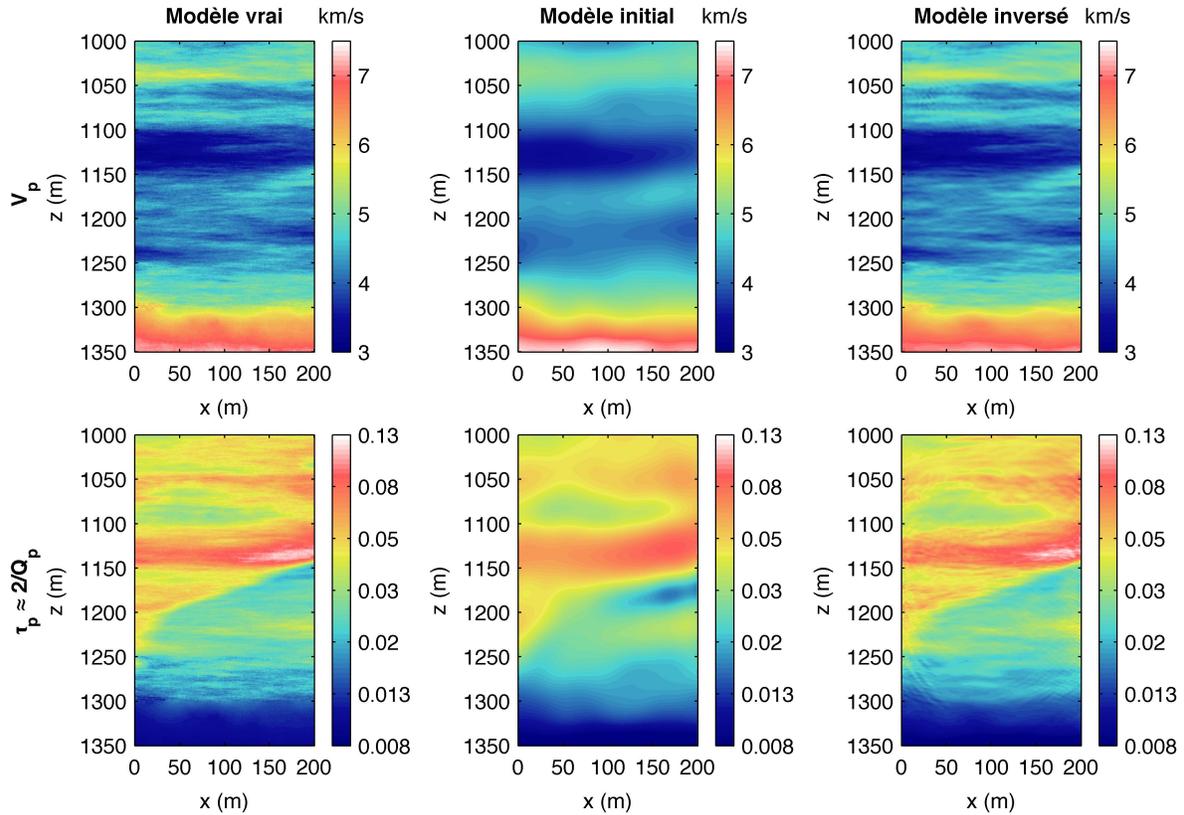


Figure 5.11 – Le modèle après l’injection de CO_2 : modèle vrai (première colonne), modèle initial (deuxième colonne) et modèle inversé (troisième colonne). La vitesse des ondes P est présentée à la première rangée et le niveau d’atténuation est présenté à la deuxième rangée.

du niveau d’atténuation sont mieux reconstruits que pour le modèle avant injection. La Figure 5.13 confirme que les valeurs extrêmes sont mieux reproduites pour ce modèle. Les anomalies dues au couplage sont plus faibles, quoique toujours présentes. Ceci s’explique par la magnitude beaucoup plus élevée des niveaux d’atténuation suite à l’injection: comme il a été montré précédemment, le couplage s’amointrit lorsque le niveau d’atténuation est fort, sans que le gradient de V_p n’en soit trop affecté. Enfin, après inversion, les regroupements sismiques des modèles vrais et inversés sont très similaires, comme le confirme la faible amplitude des résidus (cinquième colonne de la Figure 5.12). Tout comme l’inversion avant injection, les résidus contiennent majoritairement des hautes fréquences et des ondes converties. L’inversion a donc aussi convergé pour le modèle après injection.

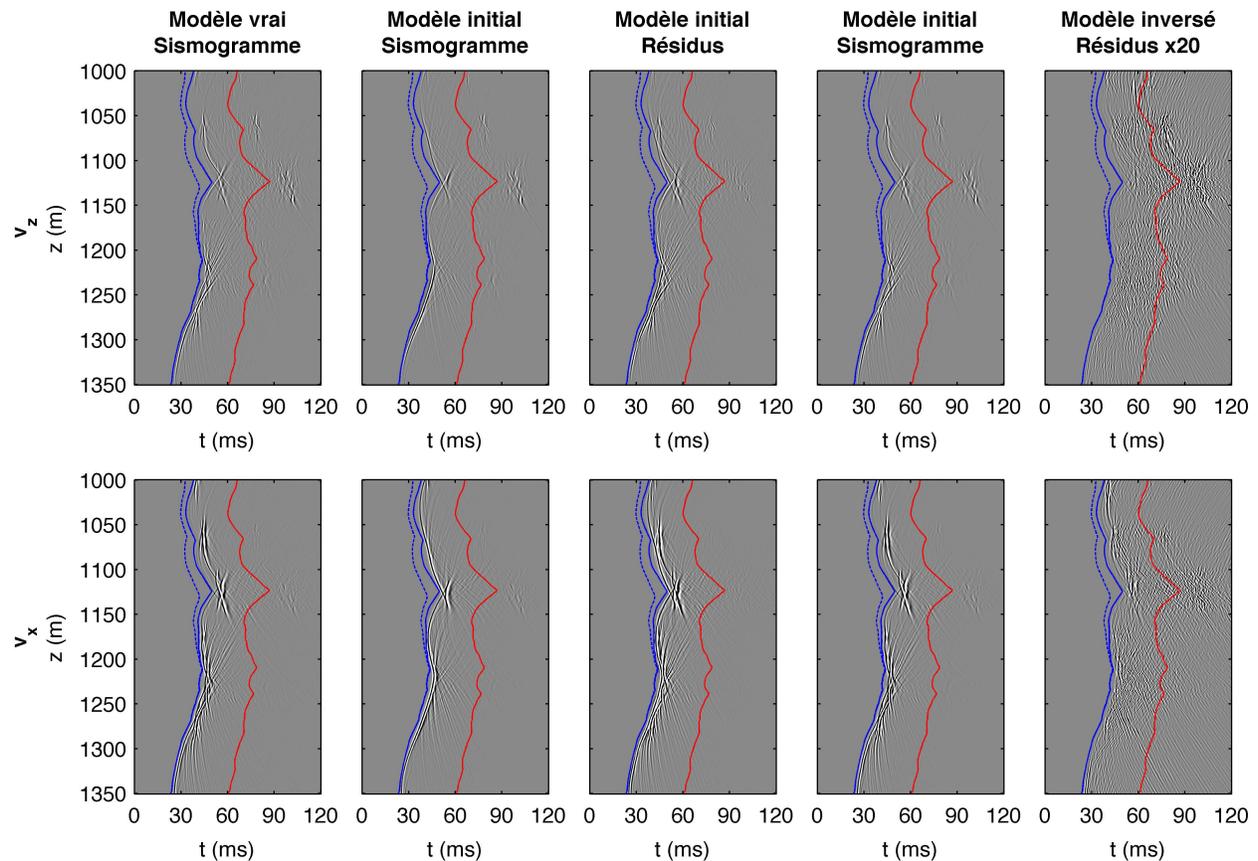


Figure 5.12 – Regroupement à déport nul pour le vrai modèle après injection (première colonne), le regroupement à déport nul et les résidus du modèle initial (deuxième et troisième colonnes) et le regroupement à déport nul et les résidus du modèle inversé (quatrième et cinquième colonnes). Les première et deuxième rangées présentent les composantes v_z et v_x de la source explosive. Les lignes bleues pleines et hachurées indiquent respectivement des premières arrivées P avant et après injection et les lignes pleines rouges indiquent les premières arrivées S .

5.3.3 Suivi du CO_2

La sensibilité de V_p et τ_p à la présence de CO_2 permet de suivre l'évolution du panache de CO_2 en suivant les changements de ces deux propriétés. Le changement des vitesses et des niveaux d'atténuation des ondes P causé par la présence de CO_2 est montré à la Figure 5.14 pour le modèle vrai (deuxième colonne). Comme ces changements ont été calculés avec le modèle de saturation partielle de White, les changements de V_p et de Q_p ne sont pas reliés linéairement à la saturation en CO_2 . La forme du panache est donc perceptible sur le changement de ces deux propriétés, quoique déformée par cette non-linéarité. Les changements des paramètres inversés sont obtenus en faisant la différence entre les modèles inversés après et avant injection. Le changement inversé de V_p (deuxième colonne de la Figure 5.14) reproduit assez précisément le changement des modèles vrais,

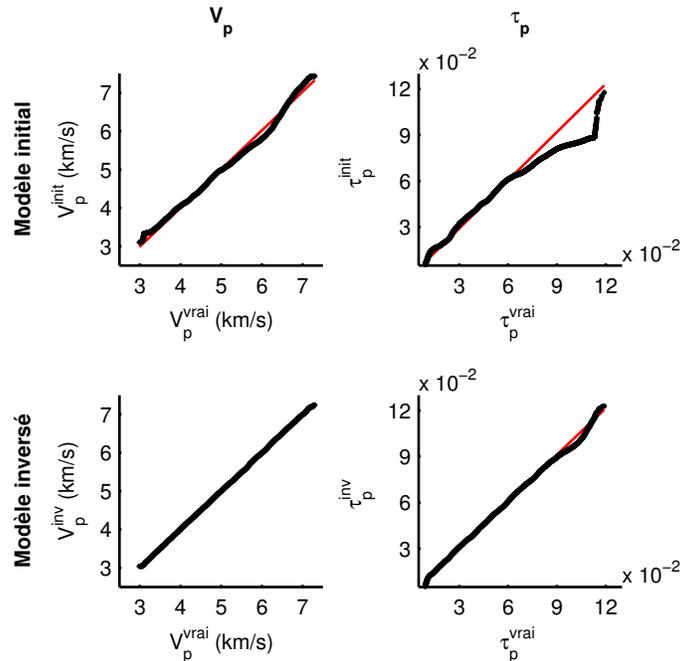


Figure 5.13 – Graphique quantile-quantile entre le modèle vrai après injection et les modèles initiaux et inversés. Les lignes rouges représentent une correspondance exacte entre les deux distributions et les points noirs représentent les quantiles calculés.

bien que quelques artefacts soient présents. Ceux-ci sont possiblement causés par la plage d’angles limitée de la configuration d’acquisition et au couplage entre les classes de paramètre. Bien que les changements relatifs de Q_p sont plus grands que pour V_p , la reconstruction de Q_p (troisième colonne de la Figure 5.14) présente plus d’artefacts. Ces artefacts sont évidemment causés par les erreurs retrouvées dans les modèles inversés de τ_p . Néanmoins, l’inversion de formes d’ondes complètes permet de retrouver la plupart des changements de grande amplitude du modèle vrai de Q_p . En bref, cette expérience synthétique montre la très grande résolution qui peut être atteinte pour le suivi temporel sismique grâce à l’inversion de formes d’ondes complètes viscoélastique, et ce, malgré une approche d’inversion en temps (*time-lapse*) très rudimentaire.

Sur ce dernier point, notons que des méthodes plus robustes d’inversion en temps permettraient de réduire significativement les artefacts d’imagerie introduits par l’inversion séparée des levés avant et après injection (Rittgers *et al.*, 2016; Hayley *et al.*, 2011; Kim *et al.*, 2009). Particulièrement, Raknes & Arntsen (2015) inversent directement pour la différence des propriétés élastiques entre les deux levés, ce qui semble être une stratégie supérieure selon Asnaashari *et al.* (2015), surtout lorsqu’un modèle a priori est utilisé. De même, l’inversion avec régularisation dans le temps semble être une approche efficace, comme le montre Maharramov & Biondi (2014) qui régularise la diffé-

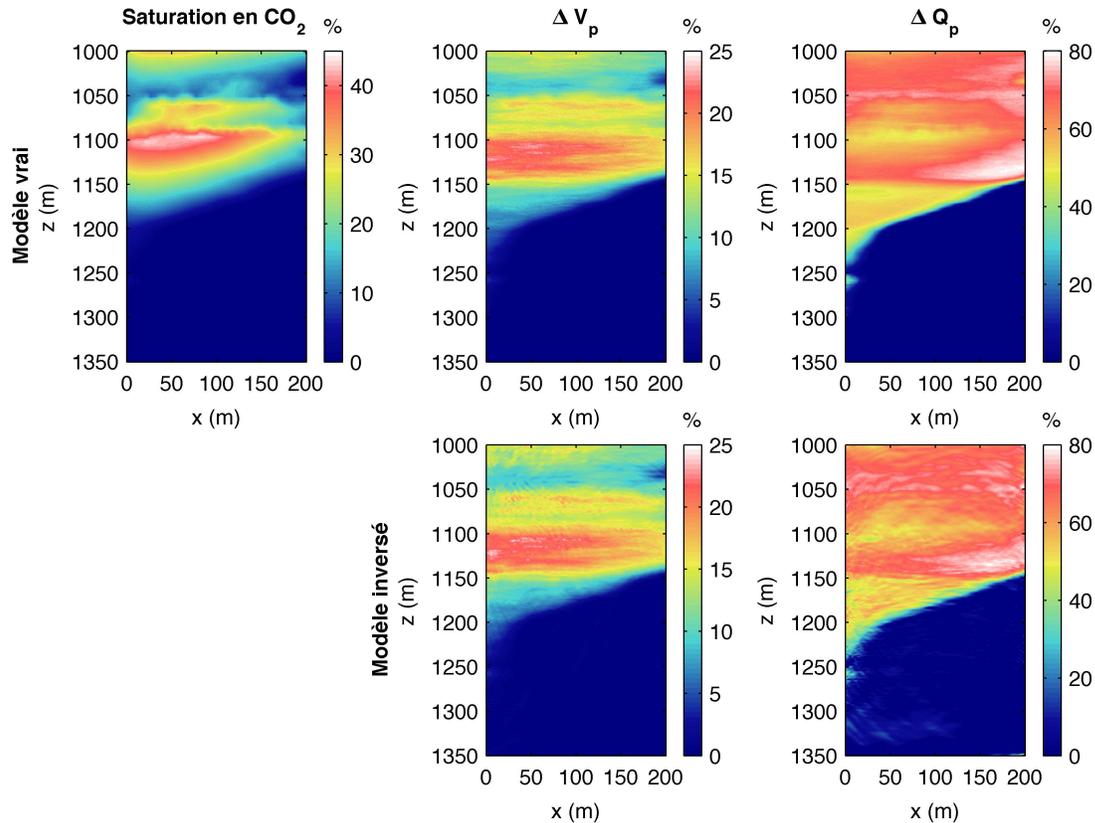


Figure 5.14 – La saturation en CO_2 après 15 ans d’injection (première colonne), le changement en V_p (deuxième colonne) et du facteur de qualité Q_p (troisième colonne) entre le modèle avant et après injection. La première rangée montre le modèle vrai et la deuxième montre les changements obtenus de l’inversion.

rence entre les deux modèles. Bref, ces approches devraient être étudiées en vue de l’application de l’inversion viscoélastique à des données réelles de suivi de l’injection du CO_2 .

5.4 Conclusion

Cet exemple synthétique montre qu’il est possible de reconstruire tant les vitesses que les facteurs de qualité avec l’inversion de formes d’ondes complètes viscoélastique. En particulier, une stratégie d’inversion hiérarchique des paramètres permet de minimiser l’effet du couplage sur les paramètres de moins grande sensibilité, dans ce cas-ci les niveaux d’atténuation. À noter que la sensibilité des données aux différentes classes de paramètre dépend du type de donnée et de la configuration d’acquisition. La hiérarchie des paramètres présentée dans ce travail n’est donc pas générale. De plus, la stratégie de sélection des données permet aussi de réduire ce couplage. Dans cet exemple,

différents types de sources ont été utilisées. Une autre stratégie serait de sélectionner différents types d'arrivées sur le même jeu de données.

La reconstruction des niveaux d'atténuation est sensible à l'amplitude de l'atténuation, et une bonne reconstruction est favorisée par de grands niveaux d'atténuation. En d'autres mots, l'empreinte des phénomènes d'atténuation doit être assez grande sur les données pour dominer les erreurs commises sur les modèles de vitesses, erreurs qui ne seront jamais nulles en pratiques. L'adoption de fonctions coût plus robustes ou l'utilisation de la Hessienne seraient deux avenues intéressantes à explorer pour diminuer la sensibilité de la reconstruction des facteurs de qualités aux vitesses sismiques.

Pour finir, l'étude de cas proposée ici montre que l'inversion de formes d'ondes complètes viscoélastique est une technologie intéressante pour suivre l'évolution du panache de CO_2 lors des activités de séquestration géologique. En effet, la résolution spatiale obtenue par celle-ci rejoint les limites physiques atteignables par les méthodes sismiques. À noter que la présente expérience s'est déroulée avec des données non bruitées et une physique parfaite. En réalité, des bruits cohérents et incohérents teinteront les données. De plus, les effets 3D, pouvant modifier la phase et l'amplitude des ondes, n'ont pas été considérés. Enfin, nous avons utilisé le modèle de White pour transformer les concentrations en CO_2 en changement de propriétés sismiques. Or, cette relation contient des paramètres qui n'ont pas été calibrés. Pour évaluer quantitativement les saturations en CO_2 à partir des changements de propriétés sismiques, ce modèle devrait être calibré et validé.

Chapitre 6

Conclusion et perspectives

L'objectif de cette thèse était de développer une approche réaliste d'inversion de formes d'ondes complètes viscoélastique dans le domaine temporel. Trois éléments essentiels d'une telle approche ont été identifiés: 1–le développement des équations adjointes de l'équation d'onde viscoélastique, 2–l'utilisation des plateformes de calcul parallèle modernes et 3–le développement d'une stratégie d'inversion permettant de reconstruire les différentes classes de paramètres viscoélastiques. Chacun de ces thèmes a été abordé dans les différents chapitres de cette thèse.

Les équations adjointes de l'équation viscoélastique anisotrope dans la formulation vitesse-contraainte ont été développées au Chapitre 3. Cette formulation de l'équation d'onde est l'une des plus répandues pour la modélisation viscoélastique dans le domaine du temps, or la littérature sur les équations adjointes de celle-ci était pratiquement inexistante. Yang *et al.* (2016b) ont tout récemment publié une revue sur le sujet. L'approche présentée dans ce document, développée parallèlement à cette dernière, en diffère quelque peu. Particulièrement, nous obtenons des équations adjointes symétriques aux équations directes. Ceci a comme avantage de pouvoir utiliser le même code de modélisation pour les deux champs. Nous avons aussi montré l'équivalence de nos résultats à la formulation du déplacement utilisée par Tarantola (1986) et Dahlen & Tromp (1998). La méthode adjointe étant au cœur de l'inversion de formes d'ondes complètes, ces résultats étaient essentiels aux étapes ultérieures de ce travail.

Le développement du code de modélisation sismique présenté au Chapitre 4 se base directement sur les équations adjointes obtenues au chapitre précédent. Ce code numérique implémente les

équations directes et adjointes et permet de calculer efficacement le gradient de la fonction coût de l'équation d'onde viscoélastique isotrope en 3D et 2D (forme P-SV et SH). Il a été montré comment la résolution en différences finies bénéficie des différents niveaux de parallélisme des plateformes de calcul modernes. En utilisant MPI et OpenCL conjointement, des accélérations de calcul substantielles peuvent être atteintes pour différents environnements de calcul hétérogènes. Cette flexibilité sur l'architecture de calcul est un avantage indéniable. La tendance étant vers des processeurs de plus en plus parallèles, celle-ci permettra de profiter des développements technologiques récents en ce domaine. Quoique l'effort de calcul de l'inversion de formes d'ondes complètes viscoélastique demeure élevé même pour les processeurs actuels, son coût devrait diminuer assez rapidement au cours des prochaines années si l'évolution des capacités de calcul suit la tendance actuelle.

Enfin, ce code a été mis à l'épreuve dans une étude de cas synthétique visant à étudier la faisabilité de l'inversion multiparamètre, notamment des paramètres d'atténuation, au Chapitre 5. Pour y parvenir, une approche minimisant le couplage entre les différentes classes de paramètres est primordiale. Dans le contexte étudié, soit le suivi sismique de l'injection du CO₂ dans des formations rocheuses en vue de sa séquestration géologique à long terme, cette approche d'inversion a permis la reconstruction de quatre classes de paramètre: les vitesses et les facteurs de qualité des ondes P et S . L'inversion des facteurs de qualité demeure toutefois ardue, principalement en raison du fort couplage entre le gradient des vitesses et celui des facteurs de qualité. La stratégie d'inversion présentée dans ce chapitre réduit ce couplage, sans y parvenir totalement. Des mesures supplémentaires devraient être adoptées pour pouvoir appliquer l'inversion viscoélastique à des vrais jeux de données, sujettes au bruit et ne répondant pas exactement à la physique de l'équation d'onde viscoélastique.

En somme, cette thèse développe, de la théorie à l'application numérique, une approche d'inversion de formes d'ondes complètes dans le domaine du temps, applicable tant à des problèmes de très grande taille en 3D qu'à des problèmes plus modestes 2D. Les résultats montrent qu'une telle approche est réaliste, tant sur le plan technique (puissance de calcul) que sur le plan pratique (plausibilité de retrouver plusieurs paramètres, dont les paramètres d'atténuation). En particulier, l'utilisation des plateformes de calculs parallèles, telles les cartes graphiques, nous laisse entrevoir qu'à court terme, nous aurons la puissance de calcul nécessaire pour appliquer l'inversion de formes d'ondes complètes multiparamètres à une gamme de problématiques beaucoup plus vaste que l'exploration pétrolière. En effet, en s'affranchissant de la barrière du coût des calculs, il serait possible

d'appliquer cette technologie révolutionnaire à une diversité d'échelles, du micromètre au kilomètre, de la biologie à la géophysique planétaire.

Les résultats de cette thèse ne sont que les premiers pas vers cette visée plus générale. Plusieurs écueils demeurent, notamment en raison de la non-linéarité de l'inversion de formes d'ondes complètes. Ce thème, peu abordé dans ce travail, empêche l'application robuste de l'inversion de formes d'ondes complètes de façon autonome, et pour l'instant, un modèle initial assez précis doit être bâti préalablement à l'inversion. À noter que plusieurs travaux en cours sur cette question laissent espérer que cette contrainte sera levée dans un avenir proche, par exemple avec la formulation de nouvelles fonctions coûts. Des stratégies d'inversions multiparamètres plus robustes doivent aussi être mises au point, par exemple avec l'utilisation de la Hessienne (Operto *et al.*, 2013) ou par les méthodes d'inversion d'ensemble (Dietterich, 2000; Ren *et al.*, 2016). Enfin, une modélisation physique plus précise devrait être adoptée pour certaines applications: par exemple, une inversion anisotrope serait avantageuse dans certains milieux sédimentaires. Pour conclure, l'intense développement des dernières années sur l'inversion de formes d'ondes complètes laisse présager que ces problématiques seront surmontées à moyen terme, permettant à cette technologie de devenir un outil de traitement quantitatif général applicable dans une diversité de contextes.

Références

- Abdelkhalek R, Calandra H, Coulaud O, Roman J & Latu G (2009). Fast seismic modeling and reverse time migration on a GPU cluster. *High Performance Computing & Simulation, 2009. HPCS'09. International Conference on*, IEEE, pages 36–43.
- Ajo-Franklin JB, Peterson J, Doetsch J & Daley TM (2013). High-resolution characterization of a CO₂ plume using crosswell seismic tomography: Cranfield, MS, USA. *International Journal of Greenhouse Gas Control*, 18:497–509. DOI:10.1016/j.ijggc.2012.12.018.
- Aki K & Richards PG (2002). *Quantitative Seismology*. volume 1. University Science Books.
- Alford RM, Kelly KR & Boore DM (1974). Accuracy of finite-difference modeling of the acoustic wave equation. *Geophysics*, 39(6):834–842. DOI:10.1190/1.1440470.
- Alkhalifah T & Choi Y (2012). Taming waveform inversion non-linearity through phase unwrapping of the model and objective functions. *Geophysical Journal International*, 191(3):1171–1178.
- Alkhalifah T & Choi Y (2014). From tomography to full-waveform inversion with a single objective function. *Geophysics*, 79(2):R55–R61. DOI:10.1190/geo2013-0291.1.
- Alterman Z & Karal F (1968). Propagation of elastic waves in layered media by finite difference methods. *Bulletin of the Seismological Society of America*, 58(1):367–398.
- Amdahl GM (1967). Validity of the single processor approach to achieving large scale computing capabilities. *Proceedings of the April 18-20, 1967, spring joint computer conference*, ACM, pages 483–485.
- Anderson JE, Tan L & Wang D (2012). Time-reversal checkpointing methods for RTM and FWI. *Geophysics*, 77(4):S93–S103. DOI:10.1190/geo2011-0114.1.
- Askani A, Akcelik V, Bielak J & Ghattas O (2007). Full waveform inversion for seismic velocity and anelastic losses in heterogeneous structures. *Bulletin of the Seismological Society of America*, 97(6):1990–2008. DOI:10.1785/0120070079.
- Asnaashari A, Brossier R, Garambois S, Audebert F, Thore P & Virieux J (2013). Regularized seismic full waveform inversion with prior model information. *Geophysics*, 78(2):R25–R36. DOI:10.1190/geo2012-0104.1.
- Asnaashari A, Brossier R, Garambois S, Audebert F, Thore P & Virieux J (2015). Time-lapse seismic imaging using regularized full-waveform inversion with a prior model: which strategy? *Geophysical Prospecting*, 63(1):78–98. DOI:10.1111/1365-2478.12176.

- Aster R, Borchers B & Thurber C (2013). *Parameter Estimation and Inverse Problems*. Academic Press, 2 edition.
- Baeten G, de Maag JW, Plessix RE, Klaassen R, Qureshi T, Kleemeyer M, Kroode Ft & Rujie Z (2013). The use of low frequencies in a full-waveform inversion and impedance inversion land seismic case study. *Geophysical Prospecting*, 61(4):701–711. DOI:10.1111/1365-2478.12010.
- Bai J, Yingst D, Bloor R & Leveille J (2014). Viscoacoustic waveform inversion of velocity structures in the time domain. *Geophysics*, 79(3):R103–R119. DOI:10.1190/geo2013-0030.1.
- Barton N (2007). *Rock Quality, Seismic Velocity, Attenuation and Anisotropy*. CRC press.
- Baysal E, Kosloff DD & Sherwood JWC (1983). Reverse time migration. *Geophysics*, 48(11):1514–1524. DOI:10.1190/1.1441434.
- Ben-Hadj-Ali H, Operto S & Virieux J (2008). Velocity model building by 3D frequency-domain, full-waveform inversion of wide-aperture seismic data. *Geophysics*, 73(5):VE101–VE117. DOI:10.1190/1.2957948.
- Ben-Menahem A & Singh SJ (2012). *Seismic Waves and Sources*. Springer Science & Business Media.
- Bieniawski ZT (1989). *Engineering rock mass classifications: a complete manual for engineers and geologists in mining, civil, and petroleum engineering*. John Wiley & Sons.
- Blanch JO, Robertsson JOA & Symes WW (1995). Modeling of a constant Q : Methodology and algorithm for an efficient and optimally inexpensive viscoelastic technique. *Geophysics*, 60(1):176–184. DOI:10.1190/1.1443744.
- Bleibinhaus F, Hole JA, Ryberg T & Fuis GS (2007). Structure of the California Coast ranges and San Andreas Fault at SAFOD from seismic waveform inversion and reflection imaging. *Journal of Geophysical Research*, 112(B6). DOI:10.1029/2006jb004611.
- Boehm C, Hanzich M, de la Puente J & Fichtner A (2016). Wavefield compression for adjoint methods in full-waveform inversion. *Geophysics*, 81(6):R385–R397. DOI:10.1190/geo2015-0653.1.
- Bohlen T (2002). Parallel 3-D viscoelastic finite difference seismic modelling. *Computers & Geosciences*, 28(8):887–899. DOI:10.1016/s0098-3004(02)00006-7.
- Bohlen T & Saenger EH (2006). Accuracy of heterogeneous staggered-grid finite-difference modeling of Rayleigh waves. *Geophysics*, 71(4):T109–T115. DOI:10.1190/1.2213051.
- Bosch M, Mukerji T & Gonzalez EF (2010). Seismic inversion for reservoir properties combining statistical rock physics and geostatistics: A review. *Geophysics*, 75(5):75A165–75A176. DOI:10.1190/1.3478209.
- Bottou L (2010). Large-scale machine learning with stochastic gradient descent. *Proceedings of COMPSTAT'2010*, Springer, pages 177–186.
- Bouchon M, Campillo M & Gaffet S (1989). A boundary integral equation-discrete wavenumber representation method to study wave propagation in multilayered media having irregular interfaces. *Geophysics*, 54(9):1134–1140. DOI:10.1190/1.1442748.

- Bozdağ E, Trampert J & Tromp J (2011). Misfit functions for full waveform inversion based on instantaneous phase and envelope measurements. *Geophysical Journal International*, 185(2):845–870. DOI:10.1111/j.1365-246X.2011.04970.x.
- BP (2013). *BP opens new facility to house the world's largest supercomputer for commercial research*. <http://www.bp.com/en/global/corporate/press/press-releases/bp-opens-new-facility-houston-largest-supercomputer.html>.
- Brindle K (2008). New approaches for imaging tumour responses to treatment. *Nat Rev Cancer*, 8(2):94–107. DOI:10.1038/nrc2289.
- Brossier R, Operto S & Virieux J (2009). Seismic imaging of complex onshore structures by 2D elastic frequency-domain full-waveform inversion. *Geophysics*, 74(6):WCC105–WCC118. DOI:10.1190/1.3215771.
- Brown AR (1996). *Interpretation of Three-Dimensional Seismic Data*. Published jointly by American Association of Petroleum Geologists and the Society of Exploration Geophysicists.
- Bunks C, Saleck FM, Zaleski S & Chavent G (1995). Multiscale seismic waveform inversion. *Geophysics*, 60(5):1457–1473. DOI:10.1190/1.1443880.
- Carcione JM (1990). Wave propagation in anisotropic linear viscoelastic media: theory and simulated wavefields. *Geophysical Journal International*, 101(3):739–750. DOI:10.1111/j.1365-246X.1990.tb05580.x.
- Carcione JM, Kosloff D & Kosloff R (1988a). Viscoacoustic wave propagation simulation in the earth. *Geophysics*, 53(6):769–777. DOI:10.1190/1.1442512.
- Carcione JM, Kosloff D & Kosloff R (1988b). Wave propagation simulation in a linear viscoelastic medium. *Geophysical Journal International*, 95(3):597–611. DOI:10.1111/j.1365-246X.1988.tb06706.x.
- Carcione JM, Picotti S, Gei D & Rossi G (2006). Physics and seismic modeling for monitoring CO₂ storage. *Pure and Applied Geophysics*, 163(1):175–207. DOI:10.1007/s00024-005-0002-1.
- Castellanos C, Etienne V, Hu G, Operto S, Brossier R & Virieux J (2011). Algorithmic and methodological developments towards full waveform inversion in 3D elastic media. *Expanded Abstracts of 81st Annual Internat SEG Mtg*, pages 2793–2798.
- Casula G & Carcione J (1992). Generalized mechanical model analogies of linear viscoelastic behaviour. *Bollettino di Geofisica Teorica ed Applicata*.
- Causse E, Mittet R & Ursin B (1999). Preconditioning of full-waveform inversion in viscoacoustic media. *Geophysics*, 64(1):130–145. DOI:10.1190/1.1444510.
- Charara M, Barnes C & Tarantola A (2000). Full waveform inversion of seismic data for a viscoelastic medium. *Methods and Applications of Inversion*, Springer, pages 68–81.
- Charney JG, Fjørtoft R & von Neumann J (1950). Numerical integration of the barotropic vorticity equation. *Tellus*, 2(4):237–254.
- Chavent G (1974). Identification of functional parameters in partial differential equations. *Joint Automatic Control Conference*, pages 155–156.

- Cherry J (1962). The azimuthal and polar radiation patterns obtained from a horizontal stress applied at the surface of an elastic half space. *Bulletin of the Seismological Society of America*, 52(1):27–36.
- Christensen R (2012). *Theory of Viscoelasticity: An Introduction*. Elsevier.
- Clapp RG (2008). Reverse time migration: saving the boundaries. Stanford Exploration Project.
- Courant R, Friedrichs K & Lewy H (1928). Über die partiellen differenzgleichungen der mathematischen physik. *Mathematische annalen*, 100(1):32–74.
- Crank J & Nicolson P (1947). A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Mathematical Proceedings of the Cambridge Philosophical Society*, Cambridge Univ Press, volume 43, pages 50–67.
- Dahlen F & Tromp J (1998). *Theoretical Global Seismology*. Princeton university press.
- Daley TM, Ajo-Franklin JB & Doughty C (2011). Constraining the reservoir model of an injected CO₂ plume with crosswell CASSM at the Frio-II brine pilot. *International Journal of Greenhouse Gas Control*, 5(4):1022–1030. DOI:10.1016/j.ijggc.2011.03.002.
- Daley TM & Cox D (2001). Orbital vibrator seismic source for simultaneous *p*- and *s*-wave crosswell acquisition. *Geophysics*, 66(5):1471–1480. DOI:10.1190/1.1487092.
- Daley TM, Solbau RD, Ajo-Franklin JB & Benson SM (2007). Continuous active-source seismic monitoring of CO₂ injection in a brine aquifer. *Geophysics*, 72(5):A57–A61. DOI:10.1190/1.2754716.
- de la Puente J, Ferrer M, Hanzich M, Castillo JE & Cela JM (2014). Mimetic seismic wave modeling including topography on deformed staggered grids. *Geophysics*, 79(3):T125–T141. DOI:10.1190/geo2013-0371.1.
- de la Puente J, Käser M, Dumbser M & Igel H (2007). An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - IV. Anisotropy. *Geophysical Journal International*, 169(3):1210–1228. DOI:10.1111/j.1365-246X.2007.03381.x.
- Deutsch CV & Journel AG (1992). *Gslib: Geostatistical Software Library and User's Guide*. Oxford University Press.
- Dietterich TG (2000). Ensemble methods in machine learning. *International workshop on multiple classifier systems*, volume 1857, pages 1–15.
- Dorman J & Ewing M (1962). Numerical inversion of seismic surface wave dispersion data and crust-mantle structure in the new york-pennsylvania area. *Journal of Geophysical Research*, 67(13):5227–5241.
- Du P, Weber R, Luszczek P, Tomov S, Peterson G & Dongarra J (2012). From CUDA to OpenCL: Towards a performance-portable solution for multi-platform GPU programming. *Parallel Computing*, 38(8):391–407. DOI:10.1016/j.parco.2011.10.002.
- Dussaud E, Symes WW, Williamson P, Lemaistre L, Singer P, Denel B & Cherrett A (2008). Computational strategies for reverse-time migration. *2008 SEG Annual Meeting*, Society of Exploration Geophysicists, pages 2267–2271.

- Eidsvik J, Avseth P, Omre H, Mukerji T & Mavko G (2004). Stochastic reservoir characterization using prestack seismic data. *Geophysics*, 69(4):978–993. DOI:10.1190/1.1778241.
- Emmerich H & Korn M (1987). Incorporation of attenuation into time-domain computations of seismic wave fields. *Geophysics*, 52(9):1252–1264. DOI:10.1190/1.1442386.
- Faccioli E, Maggio F, Paolucci R & Quarteroni A (1997). 2D and 3D elastic wave propagation by a pseudo-spectral domain decomposition method. *Journal of Seismology*, 1(3):237–251. DOI:10.1023/a:1009758820546.
- Fichtner A (2011). *Full Seismic Waveform Modelling and Inversion*. Springer Science & Business Media.
- Fichtner A, Bunge HP & Igel H (2006). The adjoint method in seismology - I. Theory. *Physics of the Earth and Planetary Interiors*, 157(1-2):86–104. DOI:10.1016/j.pepi.2006.03.016.
- Fichtner A & van Driel M (2014). Models and Frechet kernels for frequency-(in)dependent Q . *Geophysical Journal International*, 198(3):1878–1889. DOI:10.1093/gji/ggu228.
- Fomel S, Ying L & Song X (2013). Seismic wave extrapolation using lowrank symbol approximation. *Geophysical Prospecting*, 61(3):526–536. DOI:10.1111/j.1365-2478.2012.01064.x.
- Fornberg B (1987). The pseudospectral method: Comparisons with finite differences for the elastic wave equation. *Geophysics*, 52(4):483–501. DOI:10.1190/1.1442319.
- Fornberg B (1988). The pseudospectral method: accurate representation of interfaces in elastic wave calculations. *Geophysics*, 53(5):625–637.
- Friedlander MP & Schmidt M (2012). Hybrid deterministic-stochastic methods for data fitting. *SIAM Journal on Scientific Computing*, 34(3):A1380–A1405.
- Furse CM (2000). Faster than fourier: ultra-efficient time-to-frequency-domain conversions for FDTD simulations. *IEEE Antennas and Propagation Magazine*, 42(6):24–34. DOI:10.1109/74.894179.
- Gao L, Brossier R, Pajot B, Tago J & Virieux J (2015). An immersed free-surface boundary treatment for seismic wave simulation. *Geophysics*, 80(5):T193–T209. DOI:10.1190/geo2014-0609.1.
- Gauthier O, Virieux J & Tarantola A (1986). Two-dimensional nonlinear inversion of seismic waveforms: Numerical results. *Geophysics*, 51(7):1387–1403. DOI:10.1190/1.1442188.
- Gholami Y, Brossier R, Operto S, Ribodetti A & Virieux J (2013). Which parameterization is suitable for acoustic vertical transverse isotropic full waveform inversion? part 1: Sensitivity and trade-off analysis. *Geophysics*, 78(2):R81–R105. DOI:10.1190/geo2012-0204.1.
- Gibson J & Richard L (1994). Radiation from seismic sources in cased and cemented boreholes. *Geophysics*, 59(4):518–533. DOI:10.1190/1.1443613.
- Gosselin-Cliche B & Giroux B (2014). 3D frequency-domain finite-difference viscoelastic-wave modeling using weighted average 27-point operators with optimal coefficients. *Geophysics*, 79(3):T169–T188. DOI:10.1190/geo2013-0368.1.
- Green G (1871). *Mathematical Papers of the Late George Green: Edited by NM Ferrers*. Macmillan.

- Griewank A (1992). Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation. *Optimization Methods and Software*, 1(1):35–54. DOI:10.1080/10556789208805505.
- Griewank A & Walther A (2000). Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software*, 26(1):19–45. DOI:10.1145/347837.347846.
- Guasch L, Warner M, Nangoo T, Morgan J, Umpleby A, Stekl I & Shah N (2012). Elastic 3D full-waveform inversion. *SEG 2012 Annual Meeting*, pages 1–5.
- Hagedoorn JG (1959). The plus-minus method of interpreting seismic refraction sections. *Geophysical Prospecting*, 7(2):158–182. DOI:10.1111/j.1365-2478.1959.tb01460.x.
- Hale D (2013). Dynamic warping of seismic images. *Geophysics*, 78(2):S105–S115. DOI:10.1190/geo2012-0327.1.
- Hawkins LV (1961). The reciprocal method of routine shallow seismic refraction investigations. *Geophysics*, 26(6):806–819. DOI:10.1190/1.1438961.
- Hayley K, Pidlisecky A & Bentley LR (2011). Simultaneous time-lapse electrical resistivity inversion. *Journal of Applied Geophysics*, 75(2):401–411. DOI:10.1016/j.jappgeo.2011.06.035.
- HDF Group (2016). *The HDF format*. <https://www.hdfgroup.org>.
- Heelan PA (1953). Radiation from a cylindrical source of finite length. *Geophysics*, 18(3):685–696.
- Hestholm S (1999). Three-dimensional finite difference viscoelastic wave modelling including surface topography. *Geophysical Journal International*, 139(3):852–878. DOI:10.1046/j.1365-246x.1999.00994.x.
- Hestholm S & Ruud B (1998). 3-D finite-difference elastic wave modeling including surface topography. *Geophysics*, 63(2):613–622. DOI:10.1190/1.1444360.
- Hicks GJ (2002). Arbitrary source and receiver positioning in finite-difference schemes using kaiser windowed sinc functions. *Geophysics*, 67(1):156–165. DOI:10.1190/1.1451454.
- Holberg O (1987). Computational aspects of the choice of operator and sampling interval for numerical differentiation in large-scale simulation of wave phenomena. *Geophysical Prospecting*, 35(6):629–655. DOI:10.1111/j.1365-2478.1987.tb00841.x.
- Hu W (2016). An improved immersed boundary finite-difference method for seismic wave propagation modeling with arbitrary surface topography. *Geophysics*, 81(6):T311–T322. DOI:10.1190/geo2016-0094.1.
- Igel H, Mora P & Rioulet B (1995). Anisotropic wave propagation through finite-difference grids. *Geophysics*, 60(4):1203–1216. DOI:10.1190/1.1443849.
- Iturrarán-Viveros U & Molero M (2013). Simulation of sonic waves along a borehole in a heterogeneous formation: Accelerating 2.5-D finite differences using [py]OpenCL. *Computers & Geosciences*, 56:161–169. DOI:10.1016/j.cageo.2013.03.014.
- Jarchow CM & Thompson GA (1989). The nature of the mohorovicic discontinuity. *Annual Review of Earth and Planetary Sciences*, 17(1):475–506. DOI:10.1146/annurev.earth.17.050189.002355.

- Jiménez Tejero CE, Dagnino D, Sallarès V & Ranero CR (2015). Comparative study of objective functions to overcome noise and bandwidth limitations in full waveform inversion. *Geophysical Journal International*, 203(1):632–645. DOI:10.1093/gji/ggv288.
- Jones R (1958). In-situ measurement of the dynamic properties of soil by vibration methods. *Geotechnique*, 8(1):1–21.
- Kaeli DR, Mistry P, Schaa D & Zhang DP (2015). *Heterogeneous Computing with OpenCL 2.0*. Morgan Kaufmann.
- Kamei R & Pratt RG (2013). Inversion strategies for visco-acoustic waveform inversion. *Geophysical Journal International*, 194(2):859–884. DOI:10.1093/gji/ggt109.
- Kelly KR, Ward RW, Treitel S & Alford RM (1976). Synthetic seismograms: A finite-difference approach. *Geophysics*, 41(1):2–27. DOI:10.1190/1.1440605.
- Kim JH, Yi MJ, Park SG & Kim JG (2009). 4-d inversion of dc resistivity monitoring data acquired over a dynamically changing earth model. *Journal of Applied Geophysics*, 68(4):522–532. DOI:10.1016/j.jappgeo.2009.03.002.
- Kjartansson E (1979). Constant Q -wave propagation and attenuation. *Journal of Geophysical Research*, 84(B9):4737. DOI:10.1029/JB084iB09p04737.
- Kloc M & Danek T (2013). *The Multi GPU Accelerated Waveform Inversion in Distributed OpenCL Environment*, pages 237–244. Springer.
- Koketsu K, Fujiwara H & Ikegami Y (2004). Finite-element simulation of seismic ground motion with a voxel mesh. *Pure and Applied Geophysics*, 161(11-12). DOI:10.1007/s00024-004-2557-7.
- Komatitsch D, Barnes C & Tromp J (2000). Simulation of anisotropic wave propagation based upon a spectral element method. *Geophysics*, 65(4):1251–1260. DOI:10.1190/1.1444816.
- Komatitsch D, Erlebacher G, Göddeke D & Michéa D (2010). High-order finite-element seismic wave propagation modeling with MPI on a large GPU cluster. *Journal of Computational Physics*, 229(20):7692–7714. DOI:10.1016/j.jcp.2010.06.024.
- Komatitsch D & Martin R (2007). An unsplit convolutional perfectly matched layer improved at grazing incidence for the seismic wave equation. *Geophysics*, 72(5):SM155–SM167. DOI:10.1190/1.2757586.
- Komatitsch D & Tromp J (1999). Introduction to the spectral element method for three-dimensional seismic wave propagation. *Geophysical Journal International*, 139(3):806–822. DOI:10.1046/j.1365-246x.1999.00967.x.
- Komatitsch D, Xie Z, Bozdağ E, Sales de Andrade E, Peter D, Liu Q & Tromp J (2016). Anelastic sensitivity kernels with parsimonious storage for adjoint tomography and full waveform inversion. *Geophysical Journal International*, 206(3):1467–1478. DOI:10.1093/gji/ggw224.
- Kristek J, Moczo P & Archuleta RJ (2002). Efficient methods to simulate planar free surface in the 3D 4th-order staggered-grid finite-difference schemes. *Studia Geophysica et Geodaetica*, 46(2):355–381. DOI:10.1023/a:1019866422821.

- Kristek J, Moczo P & Galis M (2010). Stable discontinuous staggered grid in the finite-difference modelling of seismic motion. *Geophysical Journal International*, 183(3):1401–1407. DOI:10.1111/j.1365-246X.2010.04775.x.
- Käser M & Dumbser M (2006). An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - I. The two-dimensional isotropic case with external source terms. *Geophysical Journal International*, 166(2):855–877. DOI:10.1111/j.1365-246X.2006.03051.x.
- Lailly P (1983). The seismic inverse problem as a sequence of before stack migrations. *Conference on inverse scattering: theory and application*, Society for Industrial and Applied Mathematics, Philadelphia, PA, pages 206–220.
- Lamb H (1904). On the propagation of tremors over the surface of an elastic solid. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 203(359-371):1–42. DOI:10.1098/rsta.1904.0013.
- Lambaré G (2008). Stereotomography. *Geophysics*, 73(5):VE25–VE34. DOI:10.1190/1.2952039.
- Lancaster S & Whitcombe D (2000). Fast track coloured inversion. *2000 SEG Annual Meeting*, SEG.
- Lapidus L & Pinder GF (2011). *Numerical Solution of Partial Differential Equations in Science and Engineering*. John Wiley & Sons.
- Lei X & Xue Z (2009). Ultrasonic velocity and attenuation during CO₂ injection into water-saturated porous sandstone: Measurements using difference seismic tomography. *Physics of the Earth and Planetary Interiors*, 176(3-4):224–234. DOI:10.1016/j.pepi.2009.06.001.
- Levander AR (1988). Fourth-order finite-difference *P-SV* seismograms. *Geophysics*, 53(11):1425–1436. DOI:10.1190/1.1442422.
- Li X, Aravkin AY, van Leeuwen T & Herrmann FJ (2012). Fast randomized full-waveform inversion with compressive sensing. *Geophysics*, 77(3):A13–A17. DOI:10.1190/geo2011-0410.1.
- Liao O & McMechan GA (1995). 2.5D full-wavefield viscoacoustic inversion. *Geophysical Prospecting*, 43(8):1043–1059. DOI:10.1111/j.1365-2478.1995.tb00295.x.
- Liao Q & McMechan GA (1996). Multifrequency viscoacoustic modeling and inversion. *Geophysics*, 61(5):1371–1378. DOI:10.1190/1.1444060.
- Lie KA, Krogstad S, Ligaarden IS, Natvig JR, Nilsen HM & Skaflestad B (2012). Open-source MATLAB implementation of consistent discretisations on complex grids. *Computational Geosciences*, 16(2):297–322.
- Liu HP, Anderson DL & Kanamori H (1976). Velocity dispersion due to anelasticity; implications for seismology and mantle composition. *Geophysical Journal International*, 47(1):41–58.
- Love AEH (1892). *A Treatise on the Mathematical Theory of Elasticity*. volume 1. Cambridge University Press.
- Ma Y & Hale D (2013). Wave-equation reflection travelttime inversion with dynamic warping and full-waveform inversion. *Geophysics*, 78(6):R223–R233. DOI:10.1190/geo2013-0004.1.

- Maharramov M & Biondi B (2014). Joint full-waveform inversion of time-lapse seismic data sets. *SEG Technical Program Expanded Abstracts*, pages 954–959.
- Malhotra VM & Carino NJ (2003). *Handbook on Nondestructive Testing of Concrete Second Edition*. CRC press.
- Malo M & Bédard K (2012). Basin-scale assessment for CO₂ storage prospectivity in the province of Québec, Canada. *Energy Procedia*, 23:487–494.
- Marfurt KJ (1984). Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations. *Geophysics*, 49(5):533–549. DOI:10.1190/1.1441689.
- Martens J (2010). Deep learning via Hessian-free optimization. *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 735–742.
- Martens S, Kempka T, Liebscher A, Lüth S, Möller F, Myrntinen A, Norden B, Schmidt-Hattenberger C, Zimmer M, Kühn M *et al.* (2012). Europe’s longest-operating on-shore CO₂ storage site at Ketzin, Germany: a progress report after three years of injection. *Environmental Earth Sciences*, 67(2):323–334.
- Mattson TG, Sanders BA & Massingill BL (2004). *Patterns for Parallel Programming*. Pearson Education.
- Meju MA (1994). *Geophysical data analysis: understanding inverse problem theory and practice*. volume 6. Society of Exploration Geophysicists Tulsa,, OK.
- Menke W (2012). *Geophysical data analysis: Discrete inverse theory*. volume 45. Academic press.
- Meuer HW, Strohmaier E, Dongarra JJ & Simon HD (2016). Top500 supercomputer sites, edition, november 2016. *Lien: <https://www.top500.org/lists/2016/11>*.
- Meyers MA & Chawla KK (2009). *Mechanical Behavior of Materials*. volume 2. Cambridge university press Cambridge.
- Michéa D & Komatitsch D (2010). Accelerating a three-dimensional finite-difference wave propagation code using GPU graphics cards. *Geophysical Journal International*, pages no–no. DOI:10.1111/j.1365-246X.2010.04616.x.
- Miller GF & Pursey H (1954). The field and radiation impedance of mechanical radiators on the free surface of a semi-infinite isotropic solid. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 223(1155):521–541. DOI:10.1098/rspa.1954.0134.
- Moczo P (2000). 3D fourth-order staggered-grid finite-difference schemes: Stability and grid dispersion. *Bulletin of the Seismological Society of America*, 90(3):587–603. DOI:10.1785/0119990119.
- Moczo P (2002). 3D heterogeneous staggered-grid finite-difference modeling of seismic motion with volume harmonic and arithmetic averaging of elastic moduli and densities. *Bulletin of the Seismological Society of America*, 92(8):3042–3066. DOI:10.1785/0120010167.
- Moczo P (2005). On the rheological models used for time-domain methods of seismic wave propagation. *Geophysical Research Letters*, 32(1). DOI:10.1029/2004gl021598.

- Moczo P, Bystrický E, Kristek J, Carcione JM & Bouchon M (1997). Hybrid modeling of P - SV seismic motion at inhomogeneous viscoelastic topographic structures. *Bulletin of the seismological Society of America*, 87(5):1305–1323.
- Moczo P, Kristek J & Gális M (2004). Simulation of the planar free surface with near-surface lateral discontinuities in the finite-difference modeling of seismic motion. *Bulletin of the Seismological Society of America*, 94(2):760–768.
- Modrak R & Tromp J (2016). Seismic waveform inversion best practices: regional, global and exploration test cases. *Geophysical Journal International*, 206(3):1864–1889. DOI:10.1093/gji/ggw202.
- Molero M & Iturrarán-Viveros U (2013). Accelerating numerical modeling of wave propagation through 2-D anisotropic materials using OpenCL. *Ultrasonics*, 53(3):815–822.
- Mora P (1987). Nonlinear two-dimensional elastic inversion of multioffset seismic data. *Geophysics*, 52(9):1211–1228. DOI:10.1190/1.1442384.
- Mu D, Chen P & Wang L (2013). Accelerating the discontinuous Galerkin method for seismic wave propagation simulations using the graphic processing unit (GPU)—single-GPU implementation. *Computers & Geosciences*, 51:282–292.
- Métivier L, Brossier R, Mérigot Q, Oudet E & Virieux J (2016). Measuring the misfit between seismograms using an optimal transport distance: application to full waveform inversion. *Geophysical Journal International*, 205(1):345–377. DOI:10.1093/gji/ggw014.
- Métivier L, Brossier R, Virieux J & Operto S (2013). Full waveform inversion and the truncated Newton method. *SIAM Journal on Scientific Computing*, 35(2):B401–B437. DOI:10.1137/120877854.
- Müller TM, Gurevich B & Lebedev M (2010). Seismic wave attenuation and dispersion resulting from wave-induced flow in porous rocks — a review. *Geophysics*, 75(5):75A147–75A164. DOI:10.1190/1.3463417.
- Newmark NM (1959). A method of computation for structural dynamics. *Proc. ASCE*, volume 85, pages 67–94.
- Ngoc TT, Lefebvre R, Konstantinovskaya E & Malo M (2014). Characterization of deep saline aquifers in the Bécancour area, St. Lawrence Lowlands, Québec, Canada: implications for CO₂ geological storage. *Environmental earth sciences*, 72(1):119–146.
- Nguyen BD & McMechan GA (2015). Five ways to avoid storing source wavefield snapshots in 2D elastic prestack reverse time migration. *Geophysics*, 80(1):S1–S18. DOI:10.1190/geo2014-0014.1.
- Nickolls J & Dally WJ (2010). The GPU computing era. *IEEE Micro*, 30(2):56–69. DOI:10.1109/mm.2010.41.
- Njiekak G, Schmitt DR, Yam H & Kofman RS (2013). CO₂ rock physics as part of the weyburn-midale geological storage project. *International Journal of Greenhouse Gas Control*, 16:S118–S133. DOI:10.1016/j.ijggc.2013.02.007.
- Nocedal J & Wright S (2006). *Numerical Optimization*. Springer Science & Business Media.
- Nvidia (2007). Compute unified device architecture programming guide. Nvidia.

- Ohminato T & Chouet BA (1997). A free-surface boundary condition for including 3D topography in the finite-difference method. *Bulletin of the Seismological Society of America*, 87(2):494–515.
- Okamoto T, Takenaka H, Nakamura T & Aoki T (2011). Accelerating large-scale simulation of seismic wave propagation by multi-GPUs and three-dimensional domain decomposition. *Earth, Planets and Space*, 62(12):939–942. DOI:10.5047/eps.2010.11.009.
- Onishi K, Ueyama T, Matsuoka T, Nobuoka D, Saito H, Azuma H & Xue Z (2009). Application of crosswell seismic tomography using difference analysis with data normalization to monitor CO₂ flooding in an aquifer. *International Journal of Greenhouse Gas Control*, 3(3):311–321. DOI:10.1016/j.ijggc.2008.08.003.
- Operto S, Gholami Y, Prieux V, Ribodetti A, Brossier R, Metivier L & Virieux J (2013). A guided tour of multiparameter full-waveform inversion with multicomponent data: From theory to practice. *The Leading Edge*, 32(9):1040–1054. DOI:10.1190/tle32091040.1.
- Operto S, Virieux J, Amestoy P, L'Excellent JY, Giraud L & Ali HBH (2007). 3D finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: A feasibility study. *Geophysics*, 72(5):SM195–SM211. DOI:10.1190/1.2759835.
- Operto S, Virieux J, Dessa JX & Pascal G (2006). Crustal seismic imaging from multifold ocean bottom seismometer data by frequency domain full waveform tomography: Application to the eastern Nankai trough. *Journal of Geophysical Research*, 111(B9). DOI:10.1029/2005jb003835.
- Ostrander WJ (1984). Plane-wave reflection coefficients for gas sands at nonnormal angles of incidence. *Geophysics*, 49(10):1637–1648. DOI:10.1190/1.1441571.
- Palmer D (1981). An introduction to the generalized reciprocal method of seismic refraction interpretation. *Geophysics*, 46(11):1508–1518. DOI:10.1190/1.1441157.
- Park CB, Miller RD & Xia J (1999). Multichannel analysis of surface waves. *Geophysics*, 64(3):800–808. DOI:10.1190/1.1444590.
- Perozzi L, Giroux B, Kofman R & Schmitt D (2014). Preparatory work for the seismic monitoring of CO₂ storage at a prospective site in the st. lawrence lowlands, canada. *76th EAGE Conference and Exhibition 2014*.
- Perozzi L, Gloaguen E, Giroux B & Holliger K (2016). A stochastic inversion workflow for monitoring the distribution of CO₂ injected into deep saline aquifers. *Computational Geosciences*, 20(6):1287–1300.
- Pilant WL (2012). *Elastic Waves in the Earth*. volume 11. Elsevier.
- Plessix R (2009). Three-dimensional frequency-domain full-waveform inversion with an iterative solver. *Geophysics*, 74(6):WCC149–WCC157. DOI:10.1190/1.3211198.
- Plessix RE (2006). A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, 167(2):495–503. DOI:10.1111/j.1365-246X.2006.02978.x.
- Plessix RE & Cao Q (2011). A parametrization study for surface seismic full waveform inversion in an acoustic vertical transversely isotropic medium. *Geophysical Journal International*, 185(1):539–556. DOI:10.1111/j.1365-246X.2011.04957.x.

- Prasad M, Zimmer M, Berge P & Bonner B (2004). Laboratory measurements of velocity and attenuation in sediments. *University of California, Lawrence Livermore National Laboratory, Report*, 205155.
- Pratt RG (1990). Inverse theory applied to multi-source cross-hole tomography. *Geophysical Prospecting*, 38(3):311–329.
- Pratt RG (1999). Seismic waveform inversion in the frequency domain, Part 1: Theory and verification in a physical scale model. *Geophysics*, 64(3):888–901. DOI:10.1190/1.1444597.
- Pratt RG, Shin C & Hicks G (1998). Gauss-Newton and full Newton methods in frequency-space seismic waveform inversion. *Geophysical Journal International*, 133(2):341–362. DOI:10.1046/j.1365-246X.1998.00498.x.
- Prieux V, Brossier R, Operto S & Virieux J (2013). Multiparameter full waveform inversion of multicomponent ocean-bottom-cable data from the Valhall field. Part 1: Imaging compressional wave speed, density and attenuation. *Geophysical Journal International*, 194(3):1640–1664. DOI:10.1093/gji/ggt177.
- Pyrz MJ & Deutsch CV (2014). *Geostatistical Reservoir Modeling*. Oxford University Press.
- Pérez-Cota F, Smith RJ, Moradi E, Marques L, Webb KF & Clark M (2016). High resolution 3D imaging of living cells with sub-optical wavelength phonons. *Scientific Reports*, 6:39326.
- Raknes EB & Arntsen B (2015). A numerical study of 3d elastic time-lapse full-waveform inversion using multicomponent seismic data. *Geophysics*, 80(6):R303–R315. DOI:10.1190/geo2014-0472.1.
- Ravaut C, Operto S, Imbrota L, Virieux J, Herrero A & Dell’Aversana P (2004). Multiscale imaging of complex structures from multifold wide-aperture seismic data by frequency-domain full-waveform tomography: application to a thrust belt. *Geophysical Journal International*, 159(3):1032–1056. DOI:10.1111/j.1365-246X.2004.02442.x.
- Rawlinson N, Pozgay S & Fishwick S (2010). Seismic tomography: a window into deep Earth. *Physics of the Earth and Planetary Interiors*, 178(3):101–135.
- Ren Y, Zhang L & Suganthan PN (2016). Ensemble classification and regression-recent developments, applications and future directions [review article]. *IEEE Computational Intelligence Magazine*, 11(1):41–53. DOI:10.1109/mci.2015.2471235.
- Reuters (2013). *Total’s new machine beats rivals in high-tech race for oil*. <http://www.reuters.com/article/total-supercomputer-idUSL6N0CEA0F20130322>.
- Rittgers JB, Revil A, Mooney MA, Karaoulis M, Wodajo L & Hickey CJ (2016). Time-lapse joint inversion of geophysical data with automatic joint constraints and dynamic attributes. *Geophysical Journal International*, 207(3):1401–1419. DOI:10.1093/gji/ggw346.
- Riyanti CD, Erlangga YA, Plessix RE, Mulder WA, Vuik C & Oosterlee C (2006). A new iterative solver for the time-harmonic wave equation. *Geophysics*, 71(5):E57–E63. DOI:10.1190/1.2231109.
- Robertsson JOA (1996). A numerical free-surface condition for elastic/viscoelastic finite-difference modeling in the presence of topography. *Geophysics*, 61(6):1921–1934. DOI:10.1190/1.1444107.
- Robertsson JOA, Blanch JO & Symes WW (1994). Viscoelastic finite-difference modeling. *Geophysics*, 59(9):1444–1456. DOI:10.1190/1.1443701.

- Roden JA & Gedney SD (2000). Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media. *Microwave and Optical Technology Letters*, 27(5):334–339. DOI:10.1002/1098-2760(20001205)27:5<334::aid-mop14>3.0.co;2-a.
- Rubin Y & Hubbard SS (2006). *Hydrogeophysics*. volume 50. Springer Science & Business Media.
- Rubio F, Hanzich M, Farrés A, de la Puente J & María Cela J (2014). Finite-difference staggered grids in GPUs for anisotropic elastic wave propagation simulation. *Computers & Geosciences*, 70:181–189. DOI:10.1016/j.cageo.2014.06.003.
- Russell BH & Dommico S (1988). *Introduction to Seismic Inversion Methods*. volume 2. Society of Exploration Geophysicists Tulsa.
- Rutherford SR & Williams RH (1989). Amplitude-versus-offset variations in gas sands. *Geophysics*, 54(6):680–688. DOI:10.1190/1.1442696.
- Saenger EH & Bohlen T (2004). Finite-difference modeling of viscoelastic and anisotropic wave propagation using the rotated staggered grid. *Geophysics*, 69(2):583–591. DOI:10.1190/1.1707078.
- Sambridge MS, Tarantola A & Kennett BLN (1991). An alternative strategy for non-linear inversion of seismic waveforms. *Geophysical Prospecting*, 39(6):723–736. DOI:10.1111/j.1365-2478.1991.tb00341.x.
- Schiemanz A & Igel H (2013). Accelerated 3-D full-waveform inversion using simultaneously encoded sources in the time domain: application to Valhall ocean-bottom cable data. *Geophysical Journal International*, 195(3):1970–1988. DOI:10.1093/gji/ggt362.
- Schraudolph NN, Yu J & Günter S (2007). A stochastic quasi-Newton method for online convex optimization. *AISTATS*, volume 7, pages 436–443.
- Sheng J, Leeds A, Buddensiek M & Schuster GT (2006). Early arrival waveform tomography on near-surface refraction data. *Geophysics*, 71(4):U47–U57. DOI:10.1190/1.2210969.
- Shin C & Cha YH (2008). Waveform inversion in the laplace domain. *Geophysical Journal International*, 173(3):922–931. DOI:10.1111/j.1365-246X.2008.03768.x.
- Shin C & Ha W (2008). A comparison between the behavior of objective functions for waveform inversion in the frequency and laplace domains. *Geophysics*, 73(5):VE119–VE133. DOI:10.1190/1.2953978.
- Shin C, Jang S & Min DJ (2001). Improved amplitude preservation for prestack depth migration by inverse scattering theory. *Geophysical Prospecting*, 49(5):592–606. DOI:10.1046/j.1365-2478.2001.00279.x.
- Shipp RM & Singh SC (2002). Two-dimensional full wavefield inversion of wide-aperture marine seismic streamer data. *Geophysical Journal International*, 151(2):325–344. DOI:10.1046/j.1365-246X.2002.01645.x.
- Singh D & Shah PH (2005). Generalized Archie’s law for estimation of soil electrical conductivity. *Journal of ASTM International*, 2(5):1–20.
- Sirgue L, Etgen J & Albertin U (2008). 3D frequency domain waveform inversion using time domain finite difference methods. *70th EAGE Conference and Exhibition incorporating SPE EUROPEC 2008*.

- Sirgue L & Pratt RG (2004). Efficient waveform inversion and imaging: A strategy for selecting temporal frequencies. *Geophysics*, 69(1):231–248. DOI:10.1190/1.1649391.
- Sjøgren B, Øfsthus A & Sandberg J (1979). Seismic classification of rock mass qualities. *Geophysical Prospecting*, 27(2):409–442.
- Skjervheim JA, Evensen G, Aanonsen SI, Ruud BO & Johansen TA (2005). Incorporating 4D seismic data in reservoir simulation models using ensemble Kalman filter. *SPE Annual Technical Conference and Exhibition*, Society of Petroleum Engineers.
- Song Z, Williamson PR & Pratt RG (1995). Frequency-domain acoustic-wave modeling and inversion of crosshole data: Part II-Inversion method, synthetic experiments and real-data results. *Geophysics*, 60(3):796–809. DOI:10.1190/1.1443818.
- Spetzler J, Xue Z, Saito H & Nishizawa O (2008). Case story: time-lapse seismic crosswell monitoring of CO₂ injected in an onshore sandstone aquifer. *Geophysical Journal International*, 172(1):214–225. DOI:10.1111/j.1365-246X.2007.03614.x.
- Stekl I & Pratt RG (1998). Accurate viscoelastic modeling by frequency-domain finite differences using rotated operators. *Geophysics*, 63(5):1779–1794. DOI:10.1190/1.1444472.
- Stone JE, Gohara D & Shi G (2010). OpenCL: A parallel programming standard for heterogeneous computing systems. *Computing in science & engineering*, 12(1-3):66–73.
- Strick E (1967). The determination of Q , dynamic viscosity and transient creep curves from wave propagation measurements. *Geophysical Journal International*, 13(1-3):197–218. DOI:10.1111/j.1365-246X.1967.tb02154.x.
- Strutt JW (1885). On waves propagated along the plane surface of an elastic solid. *Proceedings of the London Mathematical Society*, 17(1):4. DOI:10.1190/1.9781560801931.ch3i.
- Sun W & Fu LY (2013). Two effective approaches to reduce data storage in reverse time migration. *Computers & Geosciences*, 56:69–75. DOI:10.1016/j.cageo.2013.03.013.
- Symes WW (2007). Reverse time migration with optimal checkpointing. *Geophysics*, 72(5):SM213–SM221. DOI:10.1190/1.2742686.
- Tam CK & Webb JC (1993). Dispersion-relation-preserving finite difference schemes for computational acoustics. *Journal of computational physics*, 107(2):262–281.
- Tarantola A (1984a). Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, 49(8):1259–1266. DOI:10.1190/1.1441754.
- Tarantola A (1984b). Linearized inversion of seismic reflection data. *Geophysical Prospecting*, 32(6):998–1015. DOI:10.1111/j.1365-2478.1984.tb00751.x.
- Tarantola A (1986). A strategy for nonlinear elastic inversion of seismic reflection data. *Geophysics*, 51(10):1893–1903. DOI:10.1190/1.1442046.
- Tarantola A (1988). Theoretical background for the inversion of seismic waveforms including elasticity and attenuation. *Pure and Applied Geophysics*, 128(1-2):365–399.
- Tarantola A (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM.

- Toksöz MN, Cheng CH & Timur A (1976). Velocities of seismic waves in porous rocks. *Geophysics*, 41(4):621–645. DOI:10.1190/1.1440639.
- Tromp J, Tape C & Liu Q (2004). Seismic tomography, adjoint methods, time reversal and banana-doughnut kernels. *Geophysical Journal International*, 160(1):195–216. DOI:10.1111/j.1365-246X.2004.02453.x.
- van Leeuwen T, Aravkin AY & Herrmann FJ (2011). Seismic waveform inversion by stochastic optimization. *International Journal of Geophysics*, 2011:1–18. DOI:10.1155/2011/689041.
- van Leeuwen T & Herrmann FJ (2013). Fast waveform inversion without source-encoding. *Geophysical Prospecting*, 61:10–19. DOI:10.1111/j.1365-2478.2012.01096.x.
- Vigh D, Kapoor J, Moldoveanu N & Li H (2011). Breakthrough acquisition and technologies for subsalt imaging. *Geophysics*, 76(5):WB41–WB51. DOI:10.1190/geo2010-0399.1.
- Vigh D, Starr EW & Kapoor J (2009). Developing Earth models with full waveform inversion. *The Leading Edge*, 28(4):432–435. DOI:10.1190/1.3112760.
- Virieux J (1984). *sh*-wave propagation in heterogeneous media: Velocity-stress finite-difference method. *Geophysics*, 49(11):1933–1942. DOI:10.1190/1.1441605.
- Virieux J (1986). *P-SV* wave propagation in heterogeneous media: Velocity-stress finite-difference method. *Geophysics*, 51(4):889–901. DOI:10.1190/1.1442147.
- Virieux J, Calandra H & Plessix RE (2011). A review of the spectral, pseudo-spectral, finite-difference and finite-element modelling techniques for geophysical imaging. *Geophysical Prospecting*, 59(5):794–813. DOI:10.1111/j.1365-2478.2011.00967.x.
- Virieux J & Operto S (2009). An overview of full-waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26. DOI:10.1190/1.3238367.
- Voigt W (2014). *Lehrbuch der kristallphysik (mit ausschluss der kristalloptik)*. Springer-Verlag.
- Vuduc R, Chandramowliswaran A, Choi J, Guney M & Shringarpure A (2010). On the limits of GPU acceleration. *Proceedings of the 2nd USENIX conference on Hot topics in parallelism*, USENIX Association, pages 13–13.
- Weiss RM & Shragge J (2013). Solving 3D anisotropic elastic wave equations on parallel GPU devices. *Geophysics*, 78(2):F7–F15. DOI:10.1190/geo2012-0063.1.
- Weissleder R & Pittet MJ (2008). Imaging in the era of molecular oncology. *Nature*, 452(7187):580–9. DOI:10.1038/nature06917.
- White J (1975). Computed seismic speeds and attenuation in rocks with partial gas saturation. *Geophysics*, 40(2):224–232.
- Woodward MJ, Nichols D, Zdraveva O, Whitfield P & Johns T (2008). A decade of tomography. *Geophysics*, 73(5):VE5–VE11. DOI:10.1190/1.2969907.
- Yang P, Brossier R, Metivier L & Virieux J (2016a). Checkpointing-assisted reverse-forward simulation: An optimal recomputation method for FWI and RTM. *SEG Technical Program Expanded Abstracts 2016*, pages 1089–1093.

- Yang P, Brossier R, Métivier L & Virieux J (2016b). A review on the systematic formulation of 3D multiparameter full waveform inversion in viscoelastic medium. *Geophysical Journal International*, ggw262 pages. DOI:10.1093/gji/ggw262.
- Yang P, Gao J & Wang B (2014). RTM using effective boundary saving: A staggered grid GPU implementation. *Computers & Geosciences*, 68:64–72. DOI:10.1016/j.cageo.2014.04.004.
- Yilmaz O (2001). *Seismic Data Analysis*. volume 1. Society of exploration geophysicists Tulsa.
- Zahradnik J & Urban L (1984). Effect of a simple mountain range on underground seismic motion. *Geophysical Journal International*, 79(1):167–183. DOI:10.1111/j.1365-246X.1984.tb02848.x.
- Zeiler MD (2012). Adadelta: An adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.
- Zhang F, Juhlin C, Cosma C, Tryggvason A & Pratt RG (2012). Cross-well seismic waveform tomography for monitoring CO₂ injection: a case study from the Ketzin Site, Germany. *Geophysical Journal International*, 189(1):629–646. DOI:10.1111/j.1365-246X.2012.05375.x.
- Zhang J & Toksöz MN (1998). Nonlinear refraction travelttime tomography. *Geophysics*, 63(5):1726–1737. DOI:10.1190/1.1444468.
- Zhe F, Feng Q, Kaufman A & Yoakum-Stover S (2004). GPU cluster for high performance computing. *Proceedings of the 2004 ACM/IEEE conference on Supercomputing*, IEEE Computer Society, pages 47–47.
- Zimmer MA, Prasad M, Mavko G & Nur A (2007). Seismic velocities of unconsolidated sands: Part 1 — Pressure trends from 0.1 to 20 MPa. *Geophysics*, 72(1):E1–E13. DOI:10.1190/1.2399459.