



HAL
open science

La Transformée de Radon Généralisée et ses Applications à la Sismique

Hugonnet Pierre

► **To cite this version:**

Hugonnet Pierre. La Transformée de Radon Généralisée et ses Applications à la Sismique. Géophysique [physics.geo-ph]. Université de Pau et des pays de l'Adour, 1998. Français. NNT: . tel-01511030

HAL Id: tel-01511030

<https://theses.hal.science/tel-01511030>

Submitted on 20 Apr 2017

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

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

THESE

Présentée à

L'UNIVERSITE DE PAU ET DES PAYS DE L'ADOUR

**ECOLE DOCTORALE DES SCIENCES EXACTES
ET DE LEURS APPLICATIONS**

par

Pierre HUGONNET

pour obtenir le grade de

DOCTEUR

Spécialité:

Géophysique

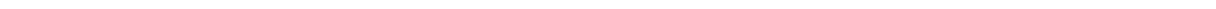
**LA TRANSFORMEE DE RADON GENERALISEE
ET SES APPLICATIONS A LA SISMIQUE**

Soutenue le 29 Janvier 1998

devant la Commission d'examen formée de:

Mme Béatrice de VOOGD, Professeur à l'UPPA Présidente
M Eric de BAZELAIRE, Expert Scientifique à Elf Exploration Production Examineur
M Guy CANADAS, Expert Scientifique à Elf Exploration Production. Examineur
M Guy CHAVENT, Professeur à l'université de Paris Dauphine Rapporteur
M Michel DIETRICH, Chargé de Recherches au CNRS Rapporteur
M Hervé PERROUD, Professeur à l'UPPA Directeur de thèse
M Simon SPITZ, Expert Scientifique à la Compagnie Générale de Géophysique. Examineur

Table des matières



Résumé	1
Avant-Propos	3
Introduction générale	5
I1 Historique et généralités sur les transformées de Radon	6
I2 Les applications des transformées de Radon en sismique	7
I3 Sujets traités dans ce rapport	8
1 Formalisme, Rappels, et Généralités	11
1.1 Introduction	12
1.2 Les familles de courbes utilisées en sismique	13
1.2.1 Cas général	13
1.2.2 Autre expression des hyperboles P-SCAN 14	
1.2.3 Comportement aux limites des équations hyperboliques 15	
1.3 Définitions	16
1.3.1 Les espaces de travail	16
1.3.2 Définition et illustration de la Transformée de Radon	16
1.3.3 Définitions de la modélisation et de la décomposition de Radon	19
1.3.4 Distributions sur D et U	20
1.4 Linéarité et continuité de la modélisation	22
1.5 Formule inverse de la modélisation pour la LRD	24
1.6 Opérateur adjoint de la modélisation	24
1.7 Inversion par les moindres carrés (décomposition en norme L2)	25
1.7.1 Définitions	25
1.7.2 Commentaires	27
1.7.3 Illustrations	27
1.8 Discrétisation	27
1.8.1 Modélisation	28
1.8.2 Opérateur adjoint de la modélisation	31
1.9 Expressions dans le domaine de Fourier	32
1.9.1 Expressions de la transformée et de la modélisation de Radon	32
1.9.2 Relation entre la Transformée de Radon Linéaire et la Transformée de Fourier 2D	
34	
1.9.3 Distributions sur D et U dans le domaine de Fourier	34
1.9.4 LRD et PRD par les moindres carrés	36
1.9.5 Opérateurs de convolution	37
1.10 Paramétrisation	42
1.11 Conclusions	44
2 Echantillonnage et Aliasing	45
2.1 Introduction	46

2.2 Echantillonnage temporel	46
2.3 Echantillonnage en p q	47
2.3.1 Ouverture p q infinie	47
2.3.2 Ouverture p q finie	48
2.3.3 Conséquences d'un sur- ou d'un sous-échantillonnage en p q	49
2.3.4 Echantillonnage pour la HRT (hyperboles P-SCAN)	52
2.4 Le pas d'échantillonnage en x - Aliasing spatial	56
2.4.1 Théorie	56
2.4.2 Approche pratique	59
2.4.3 Exemple synthétique monofréquentiel	61
2.4.4 Exemple synthétique 2	61
2.4.5 Comment remédier à l'aliasing spatial	63
2.5 Conclusions	68
3 Le Problème inverse	69
3.1 Introduction	70
3.2 Moindres carrés généralisés	70
3.3 Formalisme bayésien	72
3.4 Cadre général de résolution	75
3.5 Résolution par gradient conjugué	76
3.5.1 Problèmes quadratiques (poids constants)	76
3.5.2 Préconditionnement	78
3.5.3 Poids non constants- Problèmes non quadratiques - Algorithmes IRLS ..	78
3.6 Inversion en norme L2: propriétés	81
3.6.1 Régularisation - facteur d'amortissement	81
3.6.2 Influence du facteur d'amortissement sur l'atténuation du bruit	86
3.6.3 Considérations fréquentielles pour la LRD	88
3.7 Influence de la couverture	88
3.7.1 Pondération du modèle	88
3.7.2 Préconditionnement du gradient	91
3.8 Préconditionnement par un filtre rho	94
3.8.1 Filtres simples	94
3.8.2 filtres rho plus sophistiqués	97
3.9 Conclusions	105
4 Solutions Parcimonieuses	107
4.1 Introduction	108
4.2 Application d'une Norme Lp sur le modèle.	110
4.2.1 Présentation	110
4.2.2 Implémentation, algorithme IRLS	110
4.2.3 Exemple	113
4.2.4 Lissage du poids	113
4.2.5 Convergence - réduction du nombre d'itérations	116

4.2.6 Focalisation des évènements - propriétés anti aliasing spatial	122
4.2.7 Sensibilité au bruit - perte de certaines données	125
4.3 Application d'une Norme L_p sur les données.	130
4.3.1 Présentation	130
4.3.2 Meilleurs traitement des bruits non gaussiens	131
4.3.3 (Re-)conditionnement du problème	134
4.4 Poids sur le modèle calculé d'après les rapports Signal/Bruit	135
4.4.1 Introduction	135
4.4.2 Estimation des variances par calcul des rapports Signal/Bruit	136
4.4.3 Focalisation - Atténuation du bruit - perte de certaines données	143
4.4.4 Déplacement d'énergie dans le modèle	143
4.5 Conclusions	147
5 Décompositions de Radon avec effet AVO	149
5.1 Introduction	150
5.2 Définitions	150
5.3 Conditionnement du problème par rapport aux deux composantes	152
5.4 Décomposition en norme L_2	154
5.5 Décompositions parcimonieuses	156
5.5.1 Norme L_1 sur le modèle	156
5.5.2 Décomposition par analyse des rapports Signal/Bruit	157
5.5.3 Application (données synthétiques)	159
5.5.4 Application (données réelles)	160
5.6 Conclusions	164
6 (τ-p) 3D	165
6.1 Introduction	166
6.2 Définitions	166
6.2.1 Variables continues	166
6.2.2 Domaine de Fourier	167
6.2.3 Variables discrètes	168
6.3 Echantillonnage et aliasing	171
6.4 Décomposition en norme L_2	173
6.4.1 Domaine de Fourier, grille des données quelconque	174
6.4.2 Domaine de Fourier, grille des données pseudo-régulière	177
6.4.3 Domaine temporel, tous types de grilles	177
6.4.4 Performances comparées des différentes implémentations	182
6.5 Décompositions parcimonieuses	183
6.6 Conclusions	184
7 Rééchantillonnage et Interpolation de traces (2D/3D) par les décompositions de Radon	185

7.1 Introduction	186
7.2 Distances d'interpolation/extrapolation maximum en norme L2 (exemple sur LRD 2D)	188
7.3 Utilisation des propriétés anti-aliasing des décompositions parcimonieuses (exemple sur PRD 2D)	189
7.4 Interpolation 3D	195
7.4.1 Avantages apportés par une direction spatiale supplémentaire	195
7.4.2 Méthodologie	196
7.4.3 Description des données synthétiques de test	197
7.4.4 Rééchantillonnage de données non-aliasées	198
7.4.5 Rééchantillonnage de données aliasées	200
7.4.6 Interpolation (gaps moyens/grands)	208
7.5 Exemple sur données réelles: rééchantillonnage d'un PSV 3D	208
7.5.1 Les données	208
7.5.2 test de décimation	209
7.5.3 régularisation	214
7.5.4 Amélioration de la qualité après rééchantillonnage	219
7.6 Conclusions	221
Conclusions... et plus si affinités	223
Annexe A: le Gradient Conjugué	225
Annexe B: Convergence des algorithmes IRLS	235
Annexe C: Matrices 'Bloc-Toeplitz'	241
Annexe D: Abstract SEG Houston'95	249
Annexe E: Abstract SEG Dallas'97	255
Annexe F: Poster Vail'97	261
Nomenclature	269
Bibliographie	275

Résumé

L'objectif de cette thèse est l'étude des transformées de Radon généralisées et de certaines de leurs applications à la sismique d'exploration pétrolière. La transformée de Radon d'une image multi-dimensionnelle consiste à intégrer cette image le long de familles de courbes (pour les images 2D, ou de surfaces pour les images 3D), chaque point de la transformée étant le résultat d'une intégration. Le domaine transformé peut présenter des avantages pour appliquer certains traitements avant de revenir dans le domaine original si on connaît la transformée inverse. Le concept de décomposition de Radon, très proche de la transformée de Radon, consiste à vouloir exprimer une image comme une superposition de courbes (ou de surfaces, ou d'autres formes géométriques) affectées d'une certaine amplitude, c'est à dire comme une combinaison linéaire de courbes, les amplitudes étant les coefficients de la combinaison. Les amplitudes sont donc obtenues par la résolution d'un système linéaire.

Dans un premier temps nous étudions les propriétés des différentes transformées 2D et des différentes familles de courbes utilisées en sismique (droites, paraboles, et hyperboles). L'étude de l'échantillonnage des différentes variables du problème permet d'une part de confirmer des règles simples pour la détermination des pas d'échantillonnage et d'autre part de montrer les conséquences du sur et du sous échantillonnage. L'aliasing spatial notamment conduit à une dispersion de l'énergie dans le domaine transformé. Nous étudions également la décomposition de Radon linéaire 3D (sur des familles de plans).

Le problème de la décomposition de Radon est en général sous-déterminé, et la plupart du temps résolu par la méthode des moindres carrés. Cette méthode simple et efficace -pour ce problème- possède néanmoins des limitations importantes. Parmi celles-ci l'impossibilité de traiter correctement des données spatialement aliasées, qui sont pourtant courantes en sismiques. Les décompositions parcimonieuses, qui reposent sur le principe de décomposer l'image en utilisant le moins de courbes possible, apportent des réponses à ces limitations. A leur tour ces méthodes présentent toutefois des inconvénients, notamment une grande sensibilité à l'estimation du niveau de bruit présent sur les données, ou encore un coût important en temps de calcul. Le problème de l'estimation du niveau de bruit peut être résolu par une estimation automatique et locale des rapports signal/bruit. Le gradient conjugué est la méthode d'optimisation idéale pour toutes ces décompositions (adapté dans le cadre des algorithmes IRLS (Iteratively Reweighted Least Squares) pour les décompositions parcimonieuses).

Dans un deuxième temps nous utilisons les décompositions de Radon pour des applications d'interpolation et de rééchantillonnage de traces suivant deux directions spatiales, les décompositions pouvant en effet s'appliquer à des données de géométries quelconques, régulières ou irrégulières. L'aliasing spatial impose l'utilisation des décompositions parcimonieuses. Ces dernières permettent d'interpoler des traces au travers de gaps dont la taille peut atteindre 6 fois le pas d'échantillonnage théorique. Des gaps plus grands nécessitent l'utilisation d'autres techniques que les décompositions de Radon.

Avant-Propos

Cette thèse, présentée à l'Université de Pau et des Pays de l'Adour sous la direction d'Hervé Perroud (Laboratoire de Géophysique), a été effectuée de Décembre 1994 à Janvier 1998 à la Division Recherches Géophysiques et Géologiques (DRGG) de la société Elf Exploration Production, à Pau, sous la responsabilité de Guy Canadas.

Outre le fait qu'il m'a offert la possibilité de faire cette thèse, je pense avoir eu avec Guy Canadas la chance de travailler avec quelqu'un dont la compétence est reconnue par tous, qui est toujours disponible et qui a la volonté permanente de transmettre ses connaissances (tout un passé de professeur!): que demander de plus?

Je tiens par ailleurs à remercier:

la société Elf EP, qui a financé ces travaux,

Hervé Perroud qui le premier m'avait suggéré de faire une thèse, et m'a ensuite accueilli dans son équipe,

les membres du jury pour avoir accepté de juger ce travail, tout en n'ayant qu'un temps réduit pour lire le manuscrit, surtout Guy Chavent et Michel Dietrich qui en tant que rapporteurs étaient *vraiment* obligés de tout lire,

Irène Huard et Simon Spitz, de la Compagnie Générale de Géophysique, avec lesquels j'ai eu des contacts enrichissants dans le cadre d'un projet Elf/CGG sur l'interpolation de traces,

évidemment toutes les personnes qui à un moment ou à un autre m'ont apporté leur aide ou leur contribution,

toute la DRGG pour l'excellente ambiance qui y régnait,

le Béarn pour avoir eu la bonne idée d'être à la fois à la montagne, et à la mer (en annexant le Pays Basque, bien sûr) et au soleil (euh... et un peu de pluie pour pluie pour être tout à fait honnête),

l'humanité toute entière, comme il est d'usage...

Massy, le 30 Novembre 1998

Introduction générale

II - Historique et généralités sur les transformées de Radon

Les bases mathématiques de la **transformée de Radon** (qui est une **transformation intégrale** dans des espaces multi-dimensionnels) ont été formulées par Johann Radon en 1917 (Helgason, 1980). Aucune application physique n'est envisagée à cette époque.

A partir des années 1950, la transformée de Radon commence à trouver des applications pour des problèmes de tomographie, dans des domaines aussi variés que la médecine, l'astronomie, l'optique, les matériaux, la géophysique. Dans ces problèmes les mesures sont les résultats d'intégrations diverses, d'où l'intérêt des transformées de Radon pour en déduire les milieux traversés (par un rayon optique par exemple). En traitement d'images, une transformée très semblable (la transformée de Hough) est utilisée pour la détection ou la reconnaissance de formes.

En géophysique sismique, la transformée de Radon est parfois plus connue sous le nom de **transformée (τ, p)**, ou **slant-stack**. Elle consiste tout simplement à sommer les amplitudes d'une collection de traces sismiques (formant une image à deux dimensions: le temps t , et une variable spatiale x) le long de droites d'équations $t = \tau + p \cdot x$, et à reporter la somme pour chaque couple (τ, p) dans le plan (τ, p). Il s'agit en fait d'un cas particulier de transformée de Radon.

Les premières expérimentation de ce type ont été effectuées par Rieber (1936 & 1937). A l'époque des enregistrements analogiques, il présente une méthode qui consiste à sommer les signaux provenant de récepteurs régulièrement espacés, en appliquant un retard (analogique) différent pour chaque signal, afin de privilégier l'enregistrement des ondes arrivant avec un angle d'incidence prédéterminé. Il appelle sa méthode **CDR (Controlled Directional Reception)** et le document obtenu en balayant plusieurs angles d'incidence un **sonogramme** (chaque trace du sonogramme réclamait donc une expérimentation différente!). Un sonogramme n'est rien d'autre que la transformée (τ, p) du sismogramme que l'on aurait enregistré avec les mêmes positions récepteurs. Cette méthode a été également développée et intensivement utilisée en URSS (Riabinkin, 1957).

Plus tard, Trorey (1961) démontre que toute l'**information** contenue dans un sismogramme se retrouve dans le sonogramme sans aucune perte, éliminant ainsi la plus grosse critique à l'encontre de cette méthode. Son raisonnement passe par le domaine de Fourier.

Durrani and Bisset (1984) analysent les **propriétés** fondamentales de la Transformée de Radon.

Thorson and Claerbout (1985) formulent le **problème inverse** comme la **décomposition** d'une image en une superposition de courbes (droites, hyperboles,...) d'amplitudes constantes sous une contrainte de minimisation des résidus, et proposent plusieurs solutions pour le résoudre (opérateur adjoint, moindres carrés, solutions parcimonieuses,...). Ils remarquent notamment que l'opérateur adjoint (utilisé comme approximation de l'inverse) est la transformée de Radon utilisée habituellement.

Beylkin (1987) montre que la **Transformée de Radon Discrète (DRT)** constitue un problème à part entière, et formule son inverse exact, qui ne peut notamment pas être déduit de l'inverse de la transformée continue et infinie.

Kostov (1990) présente un algorithme d'inversion rapide dans le domaine de Fourier pour la transformée (τ, p), algorithme étendu par Darche (1990a-b) à la transformée parabolique.

Wiggins and Abma (1990), puis Zhou and Greenhalgh (1994), développent le modèle convolutif dans le domaine de Fourier pour les transformées linéaires et paraboliques, permettant d'obtenir des algorithmes d'inversion exacts ou approchés très rapides.

Gulunay (1990) compare les résultats obtenus par la transformée et la décomposition de Radon (linéaire ou parabolique), notamment du point de vue de la résolution des images obtenues.

I2 - Les applications des transformées de Radon en sismique

Les applications des transformées de Radon en sismique sont nombreuses. Elles tirent parti d'une représentation différente des données. Parmi les plus importantes, citons la décomposition en ondes planes, le filtrage des bruits aléatoires ou organisés (multiples, ground-roll,...), l'interpolation, les analyses de vitesses, la migration, et la modélisation (dans l'optique de l'inversion des données sismiques).

Chapman (1981) calcule les expressions analytiques exactes des paires de transformées pour la **décomposition en ondes planes** dans le cas d'une source ponctuelle en géométrie sphérique ou plane, et dans le cas d'une source linéaire en géométrie cylindrique. Les décompositions en ondes planes de points de tirs sont utilisées par Stoffa et al. (1981) et Treitel et al. (1982), pour le pointé de loi de vitesses notamment.

On distingue deux grands types de méthodes d'atténuation des **multiples** par les transformées de Radon:

Les premières exploitent le **NMO différentiel** entre les multiples et les réels, et sont donc adaptées aux multiples longues périodes. Hampson (1986) utilise les décompositions paraboliques sur les Collections Point Milieux corrigées de NMO afin de séparer les réels (horizontalisés après le NMO) des multiples (qui conservent une courbure résiduelle). Wiggins et Abma (1990a-b) appliquent le même type de traitement, mais après un stretch en temps $t \rightarrow t^2$ sur les CMP afin de transformer les hyperboles en paraboles. Canadas et al. (1991) et Foster and Mosher (1991) utilisent les décompositions hyperboliques sur les CMP non corrigés: le principe et les résultats sont semblables à ceux obtenus avec les décompositions paraboliques. Canadas et al. (1991), puis Hunt (1996) tirent parti des propriétés des décompositions parcimonieuses afin d'obtenir une meilleure séparation réels/multiples, même lorsque ceux-ci ont un NMO différentiel faible. Ces méthodes réclament une intervention de l'utilisateur qui doit identifier les événements comme réels ou multiples.

Les secondes exploitent la **périodicité** en temps entre un réel et ses multiples pour un mécanisme de génération donnée (les peg-legs par exemple). Cette périodicité est meilleure dans le domaine (τ, p) , permettant une suppression des multiples par déconvolution prédictive (Durrani, 1991; Lokshantov, 1993). Ces méthodes sont plus adaptées, mais non restreintes, aux multiples courtes périodes.

Zhou and Greenhalgh (1996) tentent de réunir les avantages de ces deux familles de méthodes, en combinant le domaine parabolique et un mécanisme de prédiction des multiples (basé sur l'équation des ondes et un modèle du sous-sol). Cela permet d'atténuer les multiples quel que soit leur NMO différentiel (et sans avoir à les identifier explicitement).

Les bruits cohérents linéaires de type **arrivées directes** ou **ground-roll** peuvent également être efficacement supprimé dans le domaine (τ, p) , car caractérisés par de faibles vitesses apparentes (Tatham, 1984; Hampson, 1987).

Schultz and Claerbout (1978) utilisent la transformée (τ, p) de points de tir afin d'effectuer des estimations de **vitesse**s. Cette technique permet de bien prendre en compte les fortes variations latérales de vitesses. Thorson and Claerbout (1985), Nichols (1994), Sacchi and Ulrych (1994) proposent des **analyses de vitesses** (qui classiquement consistent en une transformation de Radon hyperbolique) haute résolution, en introduisant dans l'inversion un principe de parcimonie.

Les problèmes d'**interpolation** spatiale sont liés à la prise en compte de l'**aliasing** spatial, c'est pourquoi ces deux sujets sont souvent traités ensemble. Les données (t, x) sont transformées dans le domaine (τ, p) , puis reconstruites par la transformée inverse, sur des positions x qui n'existaient pas forcément au départ.

Stoffa et al. (1981) et Lu (1985) pondèrent le slant-stack par la semblance, afin d'éliminer l'énergie induite par l'aliasing.

Maroof and Gravely (1984) proposent des choix de valeurs pour les pas d'échantillonnage Δx et Δp dans la transformée (τ, p) , en s'appuyant sur le lien entre la pente p et le nombre d'onde k .

Turner (1990), à partir de raisonnements graphiques simples, arrive aux mêmes formules. Il propose aussi d'appliquer un filtre notch pour éliminer de la transformée la partie aliasée d'un évènement de pente connue.

Lu (1989) applique également un critère de semblance sur des slant-stacks locaux afin de favoriser l'interpolation des évènements les plus cohérents dans le cas de données aliasées.

Singh et al. (1989) réduisent l'aliasing spatial en cherchant la pente dominante en tout point des données, puis en convoluant les données par des fonctions triangulaires déduites ces pentes dominantes.

Nichols (1992) impose une contrainte de continuité spectrale dans le domaine (f, p) afin d'éliminer les composantes aliasées.

Yilmaz (1994) utilise des recherches de pentes locales pour réduire l'aliasing dans la décomposition linéaire, insérées dans un processus itératif aboutissant à une vraie décomposition.

La **migration** dans le domaine (τ, p) a été abordée par Ottolini and Claerbout (1984) (sur les sections iso- p après transformée (τ, p) des CMP) et Hubral (1989) (sur la transformée (τ, p) des sections iso-offset).

Dietrich (1988) montre que la modélisation analytique des données sismiques dans un milieu stratifié 1D est beaucoup plus simple et rapide dans le domaine (τ, p) que dans le domaine (t, x) . Kormendi and Dietrich (1991) exploitent cette propriété pour inverser les données sismiques dans le domaine (τ, p) après les avoir décomposées en ondes planes.

I3 - Sujets traités dans ce rapport

Nous posons d'abord dans le chapitre 1 le problème des transformées de Radon telles qu'elles sont utilisées en sismique, afin d'en avoir une compréhension mathématique claire, aussi bien dans le domaine temps que dans le domaine de Fourier. La discrétisation des variables est discutée, ainsi que la paramétrisation des différentes familles de courbes utilisées. Les inverses en moindres carrés sont présentés.

Dans le chapitre 2 nous étudions l'échantillonnage et l'aliasing pour les différentes variables utilisées, dans les transformées de Radon linéaire et parabolique principalement, en s'attachant à décrire les conséquences en cas de sur- ou de sous-échantillonnage.

Dans le chapitre 3 le problème inverse est posé par une formulation en moindres carrés généralisés d'une part, et en incorporant le point de vue bayésien de l'inversion d'autre part. Les algorithmes d'optimisation par gradient conjugué utilisés sont décrits en détail. Dans ce chapitre sont également étudiés en détail les paramètres de l'inversion en norme L2 (moindres carrés) et diverses pondérations susceptibles d'améliorer le conditionnement du problème, notamment des 'rho-filters' optimaux chargés de compenser les effets spectraux lorsque l'on travaille dans le domaine temps.

Dans le chapitre 4 nous intéressons aux solutions parcimonieuses du problème de la décomposition de Radon. Les avantages et les inconvénients (souvent non décrits dans la littérature) de ces solutions sont mis en évidence, et pour certains de ces inconvénients des solutions sont proposées.

Dans le chapitre 5 nous proposons d'utiliser des familles de courbes dont les profils d'amplitudes sont variables (paramétrés) au lieu d'être fixés (constants en général), afin notamment de mieux modéliser les effets AVO (Amplitude Versus Offset) dans la décomposition des collections CMP. La sous-détermination intrinsèque du problème nous amène à utiliser les décompositions parcimonieuses.

Dans le chapitre 6 la décomposition de Radon linéaire 3D (τ, p_x, p_y) est étudiée en détail, aussi bien dans le domaine temps que dans le domaine de Fourier. Le cas des données irrégulières fait l'objet d'attentions particulières.

Le chapitre 7 est consacré au rééchantillonnage et à l'interpolation de traces (en 2D et en 3D) en utilisant les décompositions de Radon. Les décompositions en norme L2 permettent de rééchantillonner des données non aliasées. L'utilisation de décompositions parcimonieuses locales permet d'interpoler des données aliasées jusqu'à un certain point.

1 - Formalisme, Rappels, et Généralités

1.1 - Introduction

La transformée de Radon (RT) est mathématiquement une transformation intégrale d'une image multi-dimensionnelle (les données) dans une autre image de même dimension (le modèle).

Dans le traitement des données sismiques la forme la plus courante de RT est connue sous le nom de transformée (τ, p) , ou encore slant-stack: les collections de traces sismiques (point de tir, point milieu commun, section stack,...) peuvent être vues comme la superposition de différents événements cohérents d'une trace à l'autre et alignés (au moins localement) sur des lignes droites. La transformée (τ, p) somme les amplitudes sur une famille de lignes droites: si la droite est tangente à un événement les amplitudes se somment constructivement (en phase par rapport au signal), et destructivement dans les autres cas. Elle transforme donc les événements linéaires en des points, permettent par exemple d'effectuer des pointés ou des séparations d'ondes.

La transformée (τ, p) n'est en fait qu'un cas particulier de la transformée de Radon généralisée (GRT): la dimension des données peut être quelconque, les chemins d'intégration peuvent être des courbes (des hypersurfaces pour les dimensions sup. à 2) quelconques (hyperboles, paraboles, ellipses...), voire même tout type d'objet géométrique (pour des applications de traitement d'images), et enfin des pondérations peuvent être appliquées dans les intégrales.

Ainsi, une analyse de vitesses notamment telle qu'habituellement effectuée sur une collection CMP n'est rien d'autre qu'une transformée de Radon hyperbolique: elle consiste en effet à sommer les amplitudes le long d'hyperboles paramétrées par un temps de trajet à offset nul et une vitesse de Dix. Le pointé des maximums sur la transformée permet d'identifier les événements présents sur le CMP, avec leurs caractéristiques temps et vitesse associées. Une pondération augmentant le poids des grands offsets -plus discriminants pour les vitesses- est souvent appliquée dans l'intégration.

Les migrations de type Kirchhoff ou les opérateurs de DMO sont également des GRT. Les courbes d'intégration étant le plus souvent des hyperboles (pour la migration, ou des courbes quelconques obtenues par des calculs de temps de trajets) ou des ellipses (DMO).

Le modèle est une représentation alternative des données, en principe équivalente du point de vue de l'information (sous réserve du respect de certaines règles). Cela autorise à effectuer des traitements (filtrages) sur les modèles avant de revenir dans l'espace des données, à condition bien sûr de connaître la transformée inverse. L'expression de l'inverse pour le problème continu a été donnée par Radon dans son papier original pour certains cas de GRT (Helgason, 1980). Celui de la transformée (τ, p) notamment est bien connu (voir Chapman, 1981 ou Durrani and Bisset, 1984). L'inverse exact pour le problème discret a été formulé par Beylkin (1987).

Certains auteurs (Thorson and Claerbout, 1985), en renversant les points de vue, posent comme transformée 'directe' celle qui permet de calculer les données à partir d'un modèle. La transformée inverse consiste donc à calculer le modèle à partir des données. On constate alors que les transformations de Radon couramment utilisées sont en général les adjoints (au sens mathématique) de ces transformées directes. Ce point de vue est maintenant largement répandu et c'est celui que nous utilisons ici.

Dans ce chapitre, nous posons d'abord le problème continu, en définissant les différentes transformées utilisées (la transformée de Radon, la modélisation et son adjoint) et en étudiant leurs propriétés. Le principe d'inversion par les moindres carrés pour la modélisation est rappelé. On s'intéresse ensuite au problème discret, qui nécessite des opérations d'interpolation et d'éclatement d'échantillons. Enfin, on rappelle comment se formulent toutes ces équations dans le domaine de Fourier. Ce dernier présente certains avantages, notamment lorsque les familles de courbes utilisées sont invariantes par translation suivant le temps. Ceux-ci sont d'ordre théorique (simplification des équations), et pratique (implémentations faciles et efficaces). Néanmoins le domaine temps a d'autres avantages: prise en compte des absences de données dues aux mutes (en Fourier ces absences sont

traitées comme des échantillons nuls, ce qui n'est pas la même chose), et application de poids variant en temps. C'est pourquoi nous l'utilisons préférentiellement dans nos implémentations.

1.2 - Les familles de courbes utilisées en sismique

1.2.1 - Cas général

L'espace des données 2D est paramétré par les variables t et x (le temps et une dimension spatiale en général). On considère une famille de courbes dans cet espace, paramétrées par deux variables t_0 et x_0 . On n'utilisera ici que des courbes définissant explicitement la variable (t) en fonction de la variable (x). Une définition implicite constituerait un cas plus général, mais les familles utilisées classiquement en sismique sont toujours explicites. On note les équations des courbes utilisées sous la forme:

$$t = H(t_0, x_0, x) \quad (E1.1)$$

Dans toutes les applications usuelles, H est une fonction continue et strictement croissante de la variable t_0 . On note G la réciproque de H par rapport à t_0 :

$$t_0 = G(t, x, x_0) \quad (E1.2)$$

Parmi les familles de courbes, on peut distinguer celles qui sont **invariantes par translation** suivant un paramètre (ici t_0). Elles peuvent s'écrire sous la forme:

$$t = H(t_0, x_0, x) = t_0 + h(x_0, x) \quad (E1.3)$$

Ces familles possèdent des propriétés intéressantes, aussi bien d'un point de vue théorique, comme nous le verrons plus loin, que d'un point de vue pratique (implémentation informatique efficace).

Les familles de courbes considérées ici sont les droites (dans la célèbre transformée τ - p), les hyperboles et les paraboles. Ces deux dernières servent à analyser les collections CMP, avant ou après correction de NMO. Elles sont en général paramétrées de la façon suivante:

Famille de **droites** (τ, p): **Transformée de Radon Linéaire (LRT)**:

$$t = \tau + px \quad (E1.4)$$

Famille de **paraboles** (t, q): **Transformée de Radon Parabolique (PRT)**:

$$t = \tau + qx^2 \quad (E1.5)$$

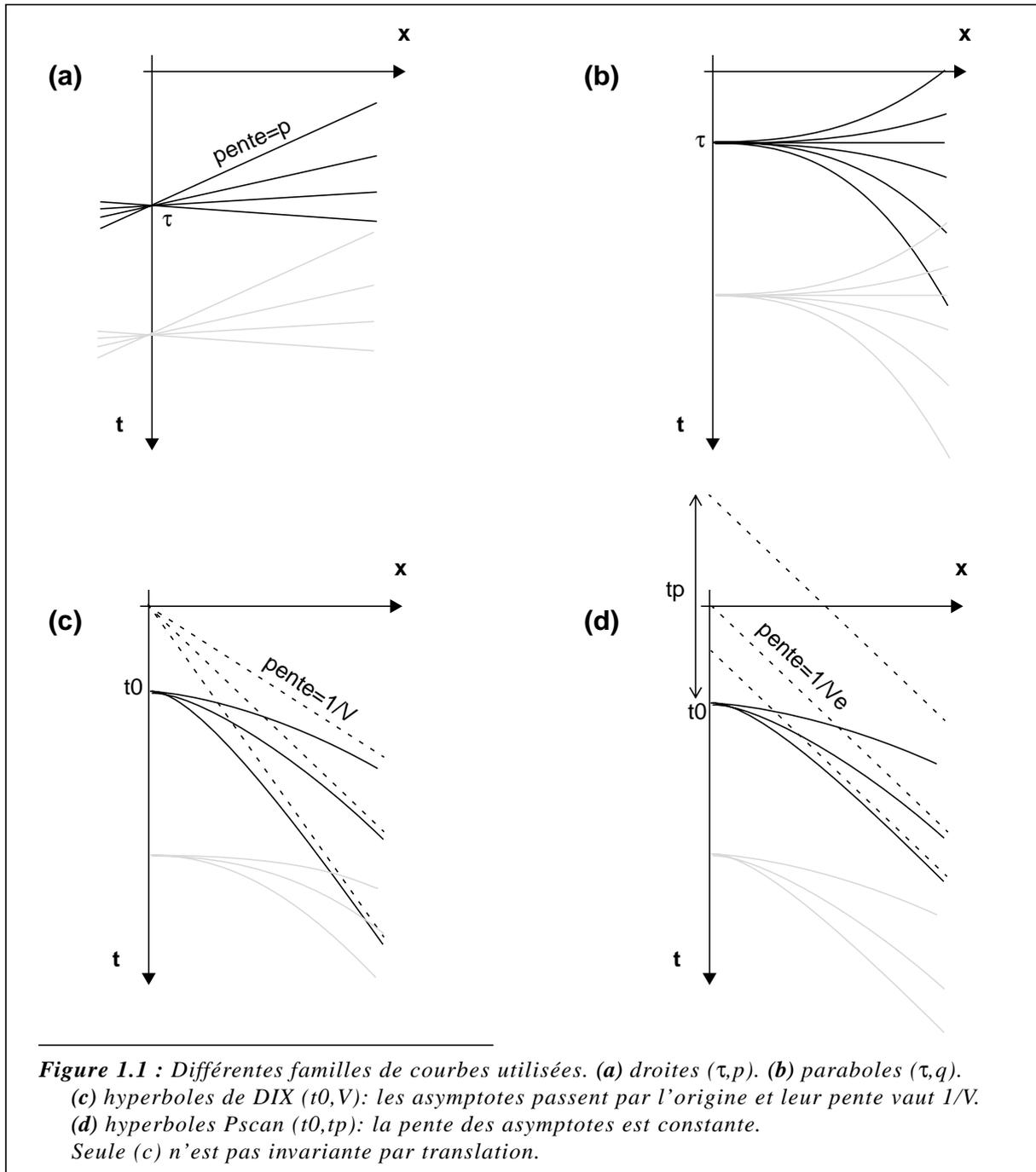
Famille des **hyperboles de DIX** ou **V-SCAN** (t_0, V): **Transformée de Radon Hyperbolique (HRT)**:

$$t = \sqrt{t_0^2 + \frac{x^2}{V^2}} \quad (E1.6)$$

Les hyperboles de DIX ne sont pas invariantes par translation suivant t . Une autre famille d'hyperboles, proposée par de Bazelaire (1988), possède cette propriété:

la famille des **hyperboles P-SCAN** (t_0, t_p) (V_e est ici une constante définissant la pente des asymptotes):

$$(t + t_p - t_0)^2 = t_p^2 + \frac{x^2}{V_e^2} \Leftrightarrow t = t_0 + t_p \left(\pm \sqrt{1 + \frac{x^2}{V_e^2 t_p^2}} - 1 \right) \quad (E1.7)$$



1.2.2 - Autre expression des hyperboles P-SCAN

On peut réécrire l'expression (E1.7) de la façon suivante:

$$t = t_0 + t_p \left(\sqrt{1 + \frac{x^2}{V_e^2 t_p^2}} - 1 \right)$$

$$= t_0 + qx^2 \frac{2}{1 + \sqrt{1 + 4V_e^2 q^2 x^2}} \quad \text{avec: } \left(q = \frac{1}{2V_e^2 t_p} \right)$$

Finalement:

$$\boxed{t = t_0 + qx^2(1 - P)}$$

$$P = V_e^2 q^2 x^2 \frac{4}{(1 + \sqrt{1 + 4V_e^2 q^2 x^2})^2} \quad q = \frac{1}{2V_e^2 t_p} \quad \text{(E1.8)}$$

L'intérêt principal de cette formulation, paramétrée par t_0 et q (V_e étant une constante), est d'englober la famille de paraboles $t = \tau + qx^2$, qui constitue le cas limite $V_e=0$ (et donc $t_p=+/-\infty$).

1.2.3 - Comportement aux limites des équations hyperboliques

Les développements limités en $x=0$ à l'ordre 4 des équations d'hyperboles (E1.6) et (E1.7) donnent:

$$\text{V-SCAN: } t = t_0 + \frac{1}{2t_0 V^2} \cdot x^2 - \frac{1}{8t_0^3 V^4} \cdot x^4 + O(x^6) \quad \text{(E1.9)}$$

$$\text{P-SCAN: } t = t_0 + \frac{1}{2t_p V_e^2} \cdot x^2 - \frac{1}{8t_p^3 V_e^4} \cdot x^4 + O(x^6) \quad \text{(E1.10)}$$

$$= t_0 + q \cdot x^2 - q^3 V_e^2 \cdot x^4 + O(x^6)$$

Le lien entre ces deux familles d'hyperboles et la famille de paraboles est obtenu par identification des termes en x^2 :

$$\boxed{q = \frac{1}{2t_0 V^2} = \frac{1}{2t_p V_e^2}} \quad \text{(E1.11)}$$

C'est donc le paramètre q qui gouverne le comportement des courbes au voisinage de 0 (on l'appellera d'une manière impropre la **courbure**).

Au voisinage de l'infini, les développements à l'ordre 1 s'écrivent:

$$\text{V-SCAN: } t = \frac{x}{V} + O\left(\frac{1}{x}\right) \quad \text{(E1.12)}$$

$$t = \frac{x}{V_e} + (t_0 - t_p) + O\left(\frac{1}{x}\right)$$

$$\text{P-SCAN: } = \frac{x}{V_e} + \left(t_0 - \frac{1}{2V_e^2 q} \right) + O\left(\frac{1}{x}\right) \quad \text{(E1.13)}$$

Dans la famille P-SCAN, toutes les hyperboles ont donc une asymptote de pente V_e , qui est le paramètre gouvernant le comportement à l'infini.

1.3 - Définitions

1.3.1 - Les espaces de travail

Le support des données est un sous-ensemble de \mathbb{R}^2 , représentant le domaine dans lequel on dispose de données enregistrées et conservées (non éliminées par un mute par exemple):

$$D = \{(t, x); T_i(x) \leq t \leq T_f(x); x_{\min} \leq x \leq x_{\max}\} \quad (E1.14)$$

Le support des modèle est également un sous-ensemble de \mathbb{R}^2 , mais choisi par l'utilisateur:

$$U = \{(t0, x0); T0_i(x0) \leq t0 \leq T0_f(x0); x0_{\min} \leq x0 \leq x0_{\max}\} \quad (E1.15)$$

Les bornes de ces deux domaines peuvent être infinies (cas purement théorique).

Soit ID l'espace des données, définies par l'ensemble des fonctions de carré intégrable sur D:

$$ID = L_2(D) \quad (E1.16)$$

On définit un produit scalaire et une norme sur cet espace:

$$\begin{aligned} \mathbf{d}_1, \mathbf{d}_2 \in ID \quad (\mathbf{d}_1, \mathbf{d}_2) &= \int_x \int_t \mathbf{d}_1(t, x) \mathbf{d}_2(t, x) dt dx \\ \|\mathbf{d}\|_{L_2} &= \sqrt{(\mathbf{d}, \mathbf{d})} \end{aligned} \quad (E1.17)$$

De même, l'espace des modèles IU est défini par l'ensemble des fonctions de carré intégrable sur U:

$$IU = L_2(U) \quad (E1.18)$$

avec pour produit scalaire et pour norme:

$$\begin{aligned} \mathbf{u}_1, \mathbf{u}_2 \in IU \quad (\mathbf{u}_1, \mathbf{u}_2) &= \int_{x0} \int_{t0} \mathbf{u}_1(t0, x0) \mathbf{u}_2(t0, x0) dt0 dx0 \\ \|\mathbf{u}\|_{L_2} &= \sqrt{(\mathbf{u}, \mathbf{u})} \end{aligned} \quad (E1.19)$$

1.3.2 - Définition et illustration de la Transformée de Radon

La **Transformée de Radon (RT)**, définie comme une transformation intégrale, associe à une fonction (\mathbf{d}) de ID une fonction (\mathbf{Rd}) de IU. La valeur de (\mathbf{Rd}) en un point $(t0, x0)$ est obtenue par intégration de

\mathbf{d} le long de la courbe $t = H(t_0, x_0, x)$. Cette application est souvent appelée en traitement sismique **sommation**, ou **stack**.

$$\mathbf{R}: \quad \text{ID} \rightarrow \text{IU}$$

$$(\mathbf{R}\mathbf{d})(t_0, x_0) = \int_x \mathbf{d}(H(t_0, x_0, x), x) dx \quad (\text{E1.20})$$

En sismique l'exemple le plus connu et utilisé de transformée de Radon est l'analyse de vitesse, qui dans sa forme la plus simple n'est rien d'autre que la transformée de Radon hyperbolique (TRH) d'une collection point milieu, basée sur la famille des hyperboles de Dix (hyperboles V-SCAN, équ. (E1.6)) (fig. 1.2)

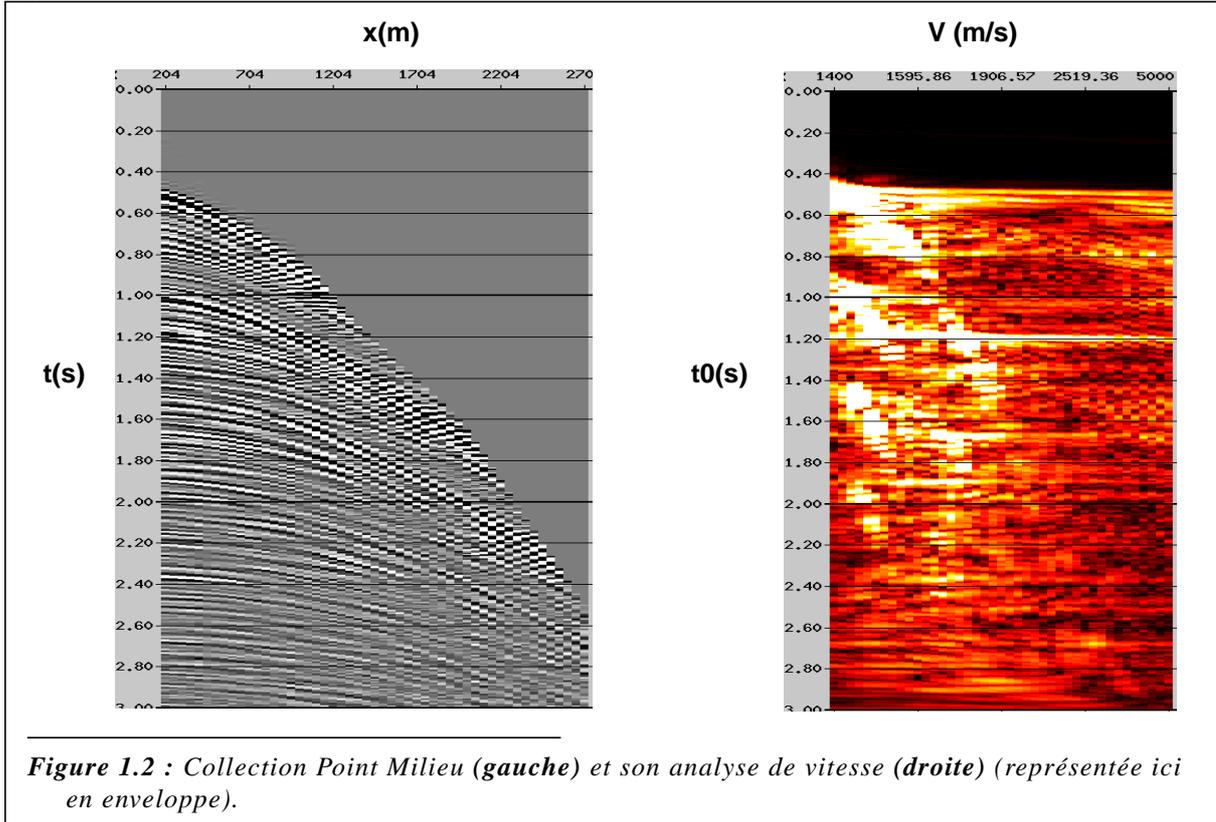


Figure 1.2 : Collection Point Milieu (*gauche*) et son analyse de vitesse (*droite*) (représentée ici en enveloppe).

L'objectif d'une transformée de Radon -linéaire par exemple- est de transformer une droite en un point, par des sommations constructives lorsque l'on utilise la bonne pente et destructives pour toute autre pente. Cet objectif est globalement atteint en pratique, mais les résultats sont en général entachés d'un certain nombre d'artefacts (fig. 1.3):

- Une focalisation imparfaite, due à l'ouverture limitée en x et à la bande passante finie du signal sismique: on obtient des sommations constructives également pour des pentes proches de la pente exacte.
- Des traînées partant du pic central, qui sont des effets de bord dus également à l'ouverture limitée en x : l'évènement n'est pas complètement 'traversé' lors de certaines sommation, qui ne sont donc pas totalement destructives.
- Un bruit diffus loin du pic central, conséquence de l'aliasing spatial (voir le chap. 2): à cause d'un échantillonnage spatial des traces insuffisant, les pentes trop fortes échantillonnent mal le signal en 'traversant' l'évènement et la sommation n'est pas destructive.
- Des données irrégulières donnent également naissance à des artefacts (défocalisation, traînées (non illustrés sur la figure)).

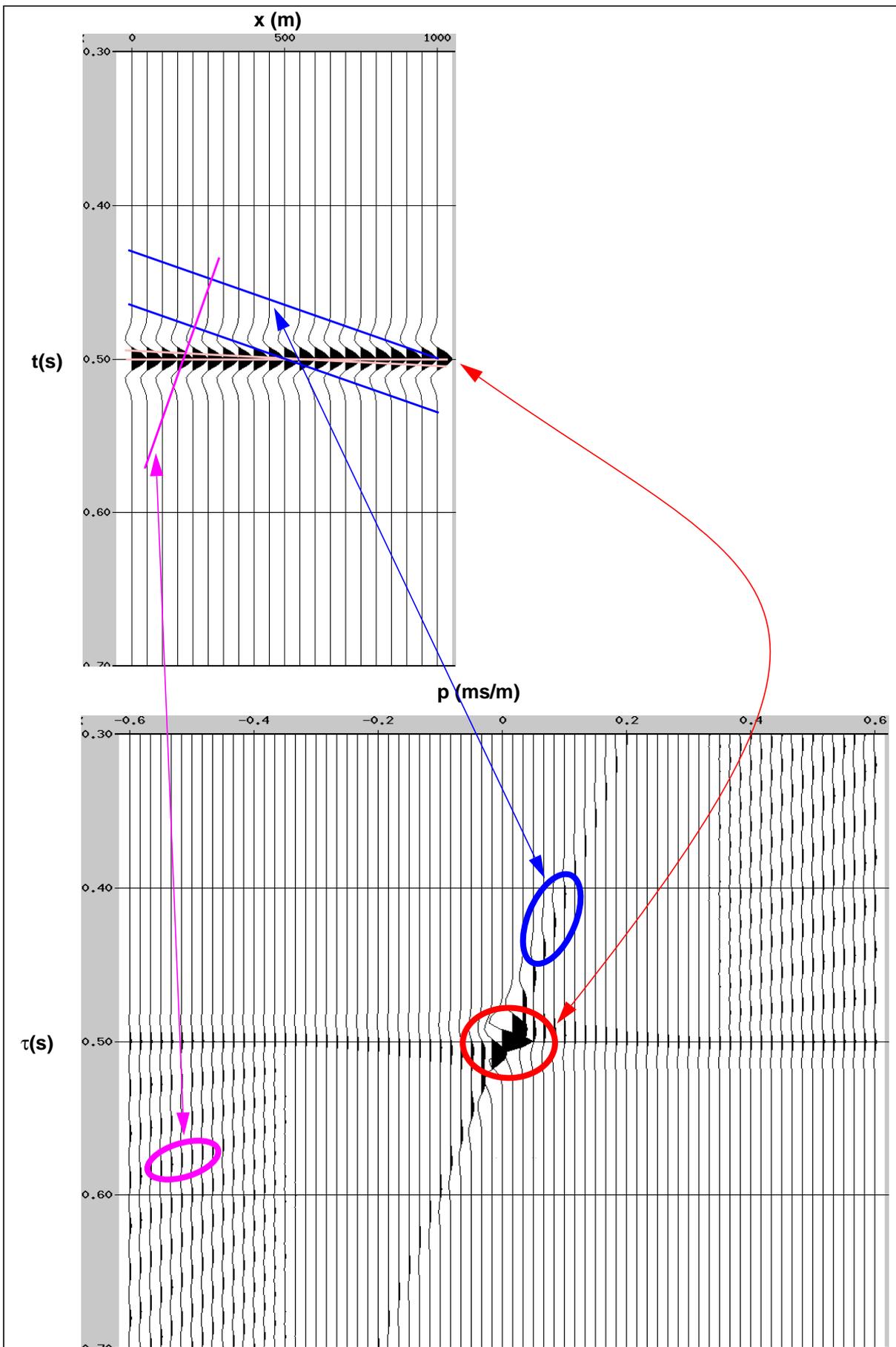


Figure 1.3 : collection de traces (*haut*) et sa LRT (*bas*): l'évènement linéaire est transformé en un point, mais plusieurs types d'artefacts sont observables...

Le plus gros inconvénient des transformées de Radon est toutefois la non-inversibilité: on ne connaît la transformée inverse (en dimension 2) que pour la transformée de Radon linéaire et pour des variables continues et infinies (cette formule d'inversion peut s'appliquer avec succès sur des données discrètes régulières dont l'ouverture est suffisante). C'est principalement pour remédier à cet inconvénient que le concept de décomposition de Radon a été introduit depuis quelques années.

1.3.3 - Définitions de la modélisation et de la décomposition de Radon

Dans le problème inverse, on définit l'application $\underline{\mathbf{L}}$, qui associe à une fonction (\mathbf{u}) de IU une fonction ($\underline{\mathbf{L}}\mathbf{u}$) de ID.

On veut que ($\underline{\mathbf{L}}\mathbf{u}$) exprime la superposition de toutes les courbes de la base, la courbe $t = H(t_0, x_0, x)$ étant affectée d'une amplitude constante $\mathbf{u}(t_0, x_0)$.

On appelle cette application **modélisation**, ou **reconstruction**.

$$\begin{array}{l} \underline{\mathbf{L}}: \quad \text{IU} \rightarrow \text{ID} \\ \underline{\mathbf{L}}\mathbf{u}(t, x) = \iint_{x_0 t_0} \mathbf{u}(t_0, x_0) \delta(t - H(t_0, x_0, x)) dt_0 dx_0 \end{array} \quad (\text{E1.21})$$

Cette notation avec une fonction de dirac est symbolique mais se justifie dans l'optique de la superposition:

Prenons une fonction test $\varphi \in \mathcal{S}(\text{D})$:

$$\begin{aligned} \langle \underline{\mathbf{L}}\mathbf{u}, \varphi \rangle &= \iint_{t \ x} \varphi(t, x) \left(\iint_{x_0 t_0} \mathbf{u}(t_0, x_0) \delta(t - H(t_0, x_0, x)) dt_0 dx_0 \right) dx dt \\ &= \iint_{x \ t_0 x_0} \mathbf{u}(t_0, x_0) \varphi(H(t_0, x_0, x), x) dx_0 dt_0 dx \end{aligned}$$

Supposons que $\mathbf{u}(t_0, x_0)$ tende vers un dirac (2D) au point (T_0, X_0) , alors:

$$\mathbf{u}(t_0, x_0) \rightarrow \delta_{2D}(t_0 - T_0, x_0 - X_0) \quad \langle \underline{\mathbf{L}}\mathbf{u}, \varphi \rangle \rightarrow \int_x \varphi(H(T_0, X_0, x), x) dx$$

$(\underline{\mathbf{L}}\mathbf{u})(t, x)$ tend donc vers une fonction de dirac 1D le long de la courbe $t = H(T_0, X_0, x)$. $\underline{\mathbf{L}}$ étant linéaire (évident), la superposition est obtenue pour toute fonction \mathbf{u} .

Nous appelons **Décomposition de Radon (RD)** la solution du problème inverse à la modélisation (nous utiliserons par la suite indifféremment les termes **inversion** et **décomposition** pour ce problème). La figure 1.4 illustre ces trois transformations de base.

Lemme: ($\underline{\mathbf{L}}\mathbf{u}$) peut s'écrire comme une transformée de Radon généralisée de IU vers ID ¹:

$$\underline{\mathbf{L}}\mathbf{u}(t, x) = \int_{x_0} \mathbf{u}(G(t, x, x_0), x_0) G_t'(t, x, x_0) dx_0 \quad (\text{E1.22})$$

¹. La famille de courbes est paramétrée par $t_0 = G(x_0, x, t)$, et la fonction de pondération est G' .

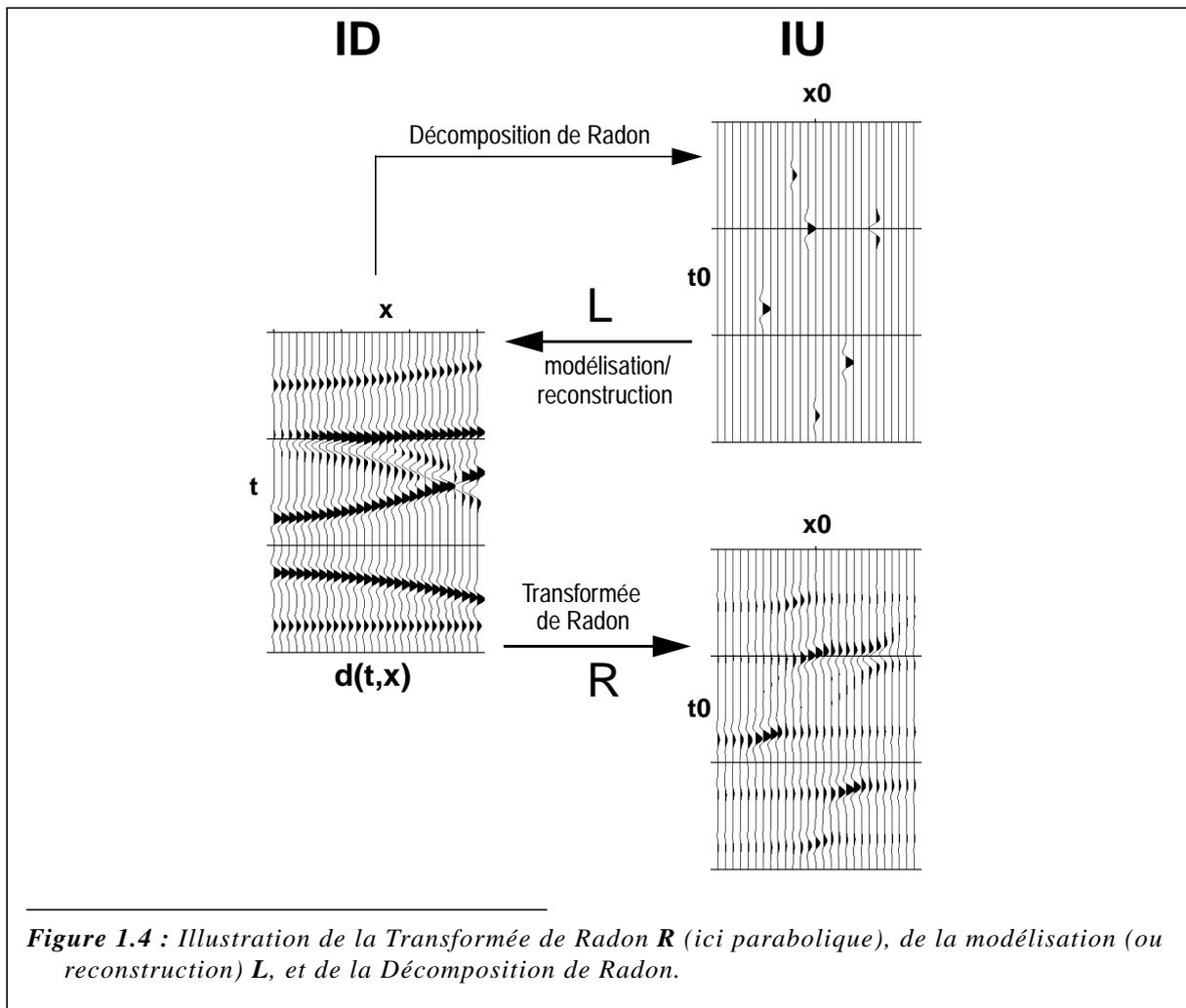


Figure 1.4 : Illustration de la Transformée de Radon R (ici parabolique), de la modélisation (ou reconstruction) L , et de la Décomposition de Radon.

Démonstration:

Posons le changement de variable $t_0 \rightarrow T$: $T = H(t_0, x_0, x)$.

Il vient: $dT = H'_{t_0}(t_0, x_0, x)dt_0 = dt_0 / G'_T(T, x, x_0)$.

$$\begin{aligned} \mathcal{L}u(t, x) &= \int_{x_0} \int_{t_0} u(t_0, x_0) \delta(t - H(t_0, x_0, x)) dt_0 dx_0 \\ &= \int_{x_0} \int_T u(G(T, x, x_0), x_0) \delta(t - T) G'_t(T, x, x_0) dT dx_0 \end{aligned}$$

D'où le résultat après intégration de la fonction de dirac.

1.3.4 - Distributions sur D et U

Rigoureusement, les définitions précédentes sont imparfaites, car d'une part les intégrales n'existent pas forcément, et d'autre part on manipule des distributions. On définit donc maintenant R et L associant à des fonctions sur D et U des distributions tempérées sur U et D:

Soit, en reprenant l'équation (E1.20):

$$\begin{aligned} \forall \varphi \in \mathcal{S}(U) \\ \langle \underline{\mathbf{R}}\mathbf{d}, \varphi \rangle &= \int_{x_0} \int_{t_0} \left(\int_x \mathbf{d}(H(t_0, x_0, x), x) dx \right) \varphi(t_0, x_0) dt_0 dx_0 \\ &= \int_{x_0} \int_{t_0} \int_x \mathbf{d}(t, x) \varphi(G(t, x, x_0), x_0) G'_t(t, x, x_0) dx dt dx_0 \end{aligned}$$

D'où la définition de $\underline{\mathbf{R}}\mathbf{d}$ en tant que distribution:

$$\boxed{\begin{aligned} \forall \varphi \in \mathcal{S}(U) \\ \langle \underline{\mathbf{R}}\mathbf{d}, \varphi \rangle &= \int_x \int_t \mathbf{d}(t, x) \left(\int_{x_0} \varphi(G(t, x, x_0), x_0) G'_t(t, x, x_0) dx_0 \right) dt dx \end{aligned}} \quad (\text{E1.23})$$

En reprenant maintenant l'équation (E1.22):

$$\begin{aligned} \forall \varphi \in \mathcal{S}(D) \\ \langle \underline{\mathbf{L}}\mathbf{u}, \varphi \rangle &= \int_x \int_t \left(\int_{x_0} \mathbf{u}(G(t, x, x_0), x_0) G'_t(t, x, x_0) dx_0 \right) \varphi(t, x) dt dx \\ &= \int_{x_0} \int_{t_0} \int_x \mathbf{u}(t_0, x_0) \varphi(H(t_0, x_0, x), x) dx dt dx_0 \end{aligned}$$

D'où la définition de $\underline{\mathbf{L}}\mathbf{u}$ en tant que distribution:

$$\boxed{\begin{aligned} \forall \varphi \in \mathcal{S}(D) \\ \langle \underline{\mathbf{L}}\mathbf{u}, \varphi \rangle &= \int_{x_0} \int_{t_0} \mathbf{u}(t_0, x_0) \left(\int_x \varphi(H(t_0, x_0, x), x) dx \right) dt_0 dx_0 \end{aligned}} \quad (\text{E1.24})$$

On vérifie facilement que l'on retombe sur les définitions classiques lorsque toutes les intégrales existent.

Ces notations inciteraient à étendre les définitions de \mathbf{R} et \mathbf{L} à des distributions, et non plus simplement à des fonctions de L_2 . En effet, on remarque que quand toutes les intégrales existent, on a:

$$\begin{aligned} \forall \varphi \in \mathcal{S}(U); \psi \in \mathcal{S}(D) \\ \langle \underline{\mathbf{R}}\mathbf{d}, \varphi \rangle &= \int_x \int_t \mathbf{d}(t, x) \left(\int_{x_0} \varphi(G(t, x, x_0), x_0) G'_t(t, x, x_0) dx_0 \right) dt dx = (\underline{\mathbf{R}}\mathbf{d}, \varphi) \\ &= (\mathbf{d}, \underline{\mathbf{L}}\varphi) = \langle \mathbf{d}, \underline{\mathbf{L}}\varphi \rangle \\ \langle \underline{\mathbf{L}}\mathbf{u}, \psi \rangle &= \int_{x_0} \int_{t_0} \mathbf{u}(t_0, x_0) \left(\int_x \psi(H(t_0, x_0, x), x) dx \right) dt_0 dx_0 = (\underline{\mathbf{L}}\mathbf{u}, \psi) \\ &= (\mathbf{u}, \underline{\mathbf{R}}\psi) = \langle \mathbf{u}, \underline{\mathbf{R}}\psi \rangle \end{aligned}$$

On voudrait donc généraliser les définitions par:

$$\boxed{\begin{aligned} \mathbf{d} \in \mathcal{S}'(D); \mathbf{u} \in \mathcal{S}'(U) \quad \forall \varphi \in \mathcal{S}(U); \psi \in \mathcal{S}(D) \\ \langle \underline{\mathbf{R}}\mathbf{d}, \varphi \rangle &= \langle \mathbf{d}, \underline{\mathbf{L}}\varphi \rangle \\ \langle \underline{\mathbf{L}}\mathbf{u}, \psi \rangle &= \langle \mathbf{u}, \underline{\mathbf{R}}\psi \rangle \end{aligned}} \quad (\text{E1.25})$$

1.4 - Linéarité et continuité de la modélisation

Pour que cela soit possible, il faudrait que $\underline{L}\varphi \in S(D)$ et $\underline{R}\psi \in S(U)$. On en trouve la démonstration dans Helgason (1980) pour la Transformation de Radon Linéaire (en dimension quelconque), mais ce résultat dépend malheureusement de la paramétrisation choisie pour les droites. On peut en effet donner le contre-exemple suivant pour la famille de droites (τ, p) (sans démonstration, mais évident):

$$\text{Soit la fonction } \varphi(t, x) = \begin{cases} \frac{1}{e^{x^2+t^2}-1} & (x^2+t^2) \leq 1 \\ 0 & (x^2+t^2) > 1 \end{cases}$$

$\varphi \in S(D)$ (voir Roddier, 1971, p.54). Considérons la famille de droites $t=\tau+px$. Les propriétés de symétrie de φ par rapport à l'origine permettent d'écrire:

$$\underline{R}\varphi(0, p) = \int_x \varphi(px, x) dx = C > 0 \quad \lim_{p \rightarrow \pm\infty} \underline{R}\varphi(\tau, p) = C$$

Cela entraîne que $\underline{R}\varphi \notin S(U)$, et par conséquent il n'est pas possible de définir ainsi la transformation de Radon d'une distribution tempérée.

Par contre, si les domaines D et U sont bornés, ces restrictions disparaissent et si $\varphi \in D(D); \psi \in D(U)$, alors $\underline{R}\varphi \in D(U); \underline{L}\psi \in D(D)$ (par définition), ce qui permet d'adopter les définitions (E1.25) pour des distributions (pas forcément tempérées) sur D et U .

Nous verrons plus loin que dans le domaine de Fourier le problème des distributions est plus facile à appréhender.

1.4 - Linéarité et continuité de la modélisation

Il est immédiat que \underline{L} est linéaire.

Dans le cas pratique des variables discrétisées et en nombres finis, la continuité de \underline{L} est acquise (application linéaire entre 2 espaces de dimensions finies). Il est néanmoins important de vérifier cette continuité dans le cas théorique des variables continues (si \underline{L} n'était pas continue, une discrétisation de plus en plus fine des domaines pourrait conduire à une dégradation des calculs).

Posons la quantité (qui est déjà apparue dans l'équ. (E1.22)):

$$S = \frac{1}{H'_{t0}(t0, x0, x)} = G'_t(t, x, x0) \quad (\text{E1.26})$$

S représente le 'stretch factor', qui apparaît habituellement lors des corrections NMO sur les CMP (fig. 1.5). La fonction H étant strictement croissante par hypothèse, le stretch factor est strictement positif ($S > 0$).

Théorème:

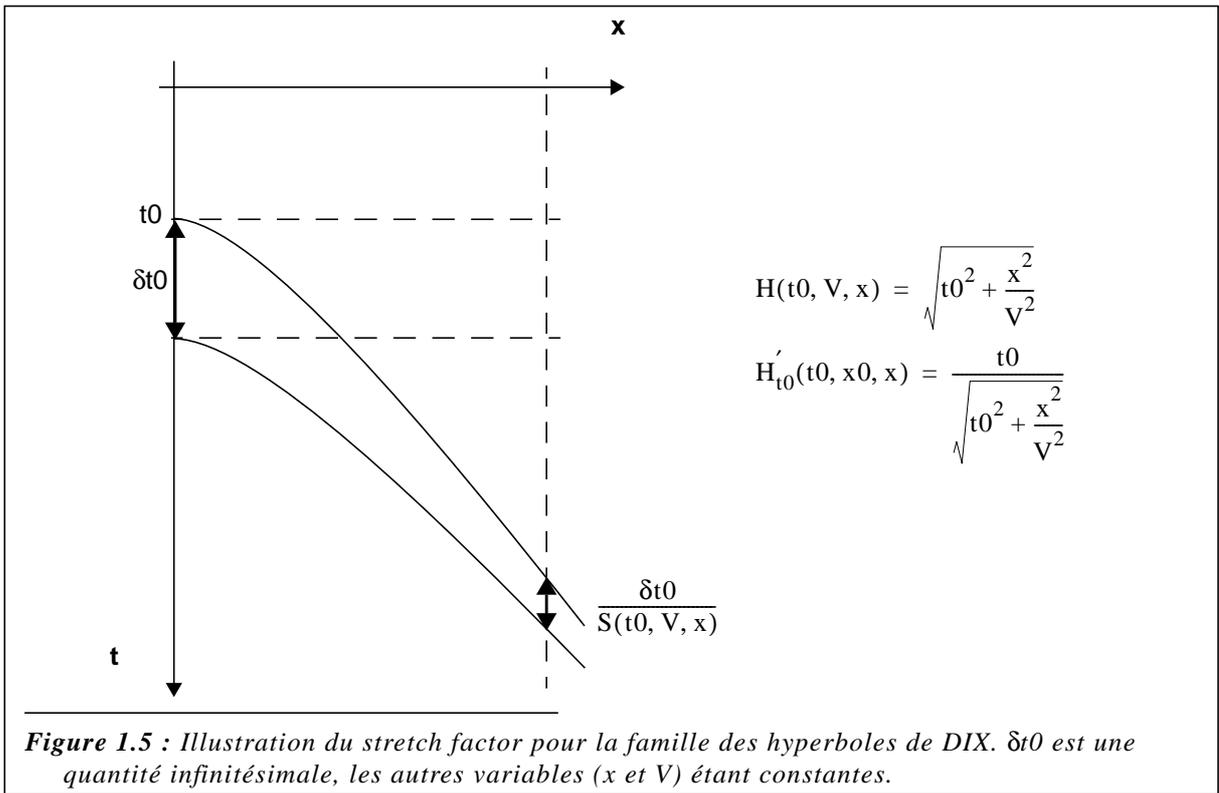
Si le stretch factor S est majoré, l'application \underline{L} de IU vers ID est continue.

Démonstration:

Prenons une fonction \mathbf{d} appartenant à $D(D)$ et calculons la quantité:

$$I = (\underline{L}\mathbf{u}, \mathbf{d}) = \iint_{x \ t} \underline{L}\mathbf{u}(t, x) \cdot \mathbf{d}(t, x) \cdot dx \cdot dt$$

On remplace $\underline{L}\mathbf{u}$ par l'expression de l'équ. (E1.22):



$$I = \iiint_{x \ t \ x_0} \mathbf{u}(G(t, x, x_0), x_0) G'_t(t, x, x_0) \mathbf{d}(t, x) dx_0 dt dx$$

$$= \iiint_{x \ t_0 \ x_0} \mathbf{u}(t_0, x_0) \mathbf{d}(H(t_0, x_0, x), x) dx_0 dt dx$$

On multiplie simplement numérateur et dénominateur par le même terme:

$$I = \iiint_{x_0 \ x \ t_0} \frac{\mathbf{u}(t_0, x_0)}{\sqrt{H'_{t_0}(t_0, x_0, x)}} \cdot \mathbf{d}(H(t_0, x_0, x), x) \cdot \sqrt{H'_{t_0}(t_0, x_0, x)} \cdot dt_0 \cdot dx \cdot dx_0$$

Par l'inégalité de Schwartz on obtient alors:

$$I \leq \iint_{x_0 \ x} \sqrt{\int_{t_0} \frac{\mathbf{u}^2(t_0, x_0)}{H'_{t_0}(t_0, x_0, x)} \cdot dt_0} \cdot \sqrt{\int_{t_0} \mathbf{d}^2(x, H(x_0, t_0, x)) \cdot H'_{t_0}(x_0, t_0, x) \cdot dt_0} \cdot dx \cdot dx_0$$

Dans la seconde racine carrée, on opère le changement de variable $t_0 \rightarrow t = H(t_0, x_0, x)$:

$$I \leq \iint_{x_0 \ x} \sqrt{\int_{t_0} \frac{\mathbf{u}^2(t_0, x_0)}{H'_{t_0}(t_0, x_0, x)} \cdot dt_0} \cdot \sqrt{\int_t \mathbf{d}^2(t, x) \cdot dt} \cdot dx \cdot dx_0$$

Par hypothèse, S est majoré ($|S| \leq M^2$). On peut donc écrire:

$$I \leq M \cdot \iint_{x_0 \ x} \sqrt{\int_{t_0} \mathbf{u}^2(t_0, x_0) \cdot dt_0} \cdot \sqrt{\int_t \mathbf{d}^2(t, x) \cdot dt} \cdot dx \cdot dx_0$$

$$\leq M \cdot \int_{x_0} \sqrt{\int_{t_0} \mathbf{u}^2(t_0, x_0) \cdot dt_0} \cdot dx_0 \cdot \int_x \sqrt{\int_t \mathbf{d}^2(t, x) \cdot dt} \cdot dx$$

En utilisant l'inégalité $(\int f)^2 \leq \int f^2$ on obtient:

$$I \leq M \cdot \sqrt{\int_{x_0} \int_{t_0} \mathbf{u}^2(t_0, x_0) \cdot dt_0 \cdot dx_0} \cdot \sqrt{\int_x \int_t \mathbf{d}^2(t, x) \cdot dt \cdot dx}$$

C'est à dire:

$$I = (\underline{\mathbf{L}}\mathbf{u}, \mathbf{d}) \leq M \cdot \|\mathbf{u}\| \cdot \|\mathbf{d}\| \quad (\text{E1.27})$$

$\underline{\mathbf{L}}$ étant linéaire, cette dernière relation suffit à montrer qu'elle est également continue, et que sa norme est majorée par M.

Concrètement, le fait que S soit majoré indique que pour x_0 fixé, les courbes obtenues en faisant varier t_0 ne font pas apparaître de point fixe: la famille des hyperboles de DIX répond à ce critère, de même que toutes les familles de courbes invariantes par translation suivant le temps (équ. (E1.3)), car dans ce cas $S=1$.

1.5 - Formule inverse de la modélisation pour la LRD

Pour le problème continu, et dans le cas où les domaines D et U sont infinis, les formules d'inversion de la modélisation (E1.21) existent, au moins pour la transformée linéaire (droites $t = \tau + px$) (démontré dans Helgason, 1980, pour des espaces de dimensions quelconques; voir aussi Chapman, 1981) (H_t est la transformée de Hilbert par rapport à la variable t (voir nomenclature)):

$$\begin{aligned} \mathbf{d}(t, x) &= \int_p \mathbf{u}(t - px, p) dp = (\underline{\mathbf{L}}\mathbf{u})(t, x) \\ &\Leftrightarrow \\ \mathbf{u}(\tau, p) &= \frac{1}{2\pi} \int_x \frac{\partial}{\partial t} H_t(\mathbf{d}(t + px, x)) dx \end{aligned} \quad (\text{E1.28})$$

Dans le domaine de Fourier, la transformée de Hilbert est un filtre en $-i \operatorname{sgn}(\omega)$, et la dérivation par rapport au temps un filtre en $i\omega$. La cascade de ces deux filtres est un filtre en $|\omega|$, d'où:

$$\mathbf{u}(\tau, p) = (F^{-1}(|\omega|))^*_{\tau} \left(\frac{1}{2\pi} \int_x \mathbf{d}(t + px, x) dx \right) = \frac{1}{2\pi} (F^{-1}(|\omega|))^*_{\tau} (\underline{\mathbf{R}}\mathbf{d}(\tau, p)) \quad (\text{E1.29})$$

Cette formule d'inversion par application du filtre en $|\omega|$ est souvent utilisée comme approximation de l'inversion pour le problème discret.

1.6 - Opérateur adjoint de la modélisation

L'opérateur adjoint $\underline{\mathbf{L}}^*$ de $\underline{\mathbf{L}}$ est une application de ID dans IU.

Théorème: Pour les produits scalaires euclidiens (équ. (E1.17) et (E1.19)), l'adjoint de $\underline{\mathbf{L}}$ est identique à l'opérateur $\underline{\mathbf{R}}$ de la Transformée de Radon, telle que définie en (E1.20):

$$\begin{aligned} &\underline{\mathbf{L}}^*: \quad \text{ID} \rightarrow \text{IU} \\ \underline{\mathbf{L}}^*\mathbf{d}(t_0, x_0) &= \int_x \mathbf{d}(H(t_0, x_0, x), x) dx = \underline{\mathbf{R}}\mathbf{d}(t_0, x_0) \end{aligned} \quad (\text{E1.30})$$

Démonstration:

Calculons la quantité:

$$\begin{aligned} (\mathbf{d}, \underline{\mathbf{L}}\mathbf{u})_D &= \int_x \int_t (\underline{\mathbf{L}}\mathbf{u})(t, x) \mathbf{d}(t, x) dx dt \\ &= \int_t \left(\int_{x_0} \mathbf{u}(G(t, x, x_0), x_0) G'_t(t, x, x_0) dx_0 \right) \mathbf{d}(t, x) dx dt \end{aligned}$$

Après le changement de variable habituel $t \rightarrow t_0 = G(t, x, x_0)$ on obtient:

$$(\mathbf{d}, \underline{\mathbf{L}}\mathbf{u})_D = \int_{x_0} \int_{t_0} \mathbf{u}(t_0, x_0) \left(\int_x \mathbf{d}(H(t_0, x_0, x), x) dx \right) dx_0 dt_0$$

En identifiant cette dernière expression à $(\underline{\mathbf{L}}^* \mathbf{d}, \mathbf{u})_U$ on obtient l'expression de l'adjoint.

1.7 - Inversion par les moindres carrés (décomposition en norme L2)

1.7.1 - Définitions

La Décomposition de Radon consiste, pour une donnée $\mathbf{d}(t,x)$ à trouver un modèle $\mathbf{u}(t_0, x_0)$ qui satisfasse l'équation de modélisation:

$$\boxed{\mathbf{d} \in ID, \text{ trouver } \mathbf{u} \in IU \ / \ \mathbf{d} = \underline{\mathbf{L}}\mathbf{u}} \quad (E1.31)$$

Ce problème est linéaire, et en travaillant avec un nombre fini de variables (cas discrétisé) il s'agit d'un système linéaire à résoudre. La solution la plus couramment utilisée (et implémentée dans l'industrie), est la résolution par les moindres carrés.

On peut obtenir cette solution en effectuant une **décomposition en valeurs singulières (SVD)** de $\underline{\mathbf{L}}$:

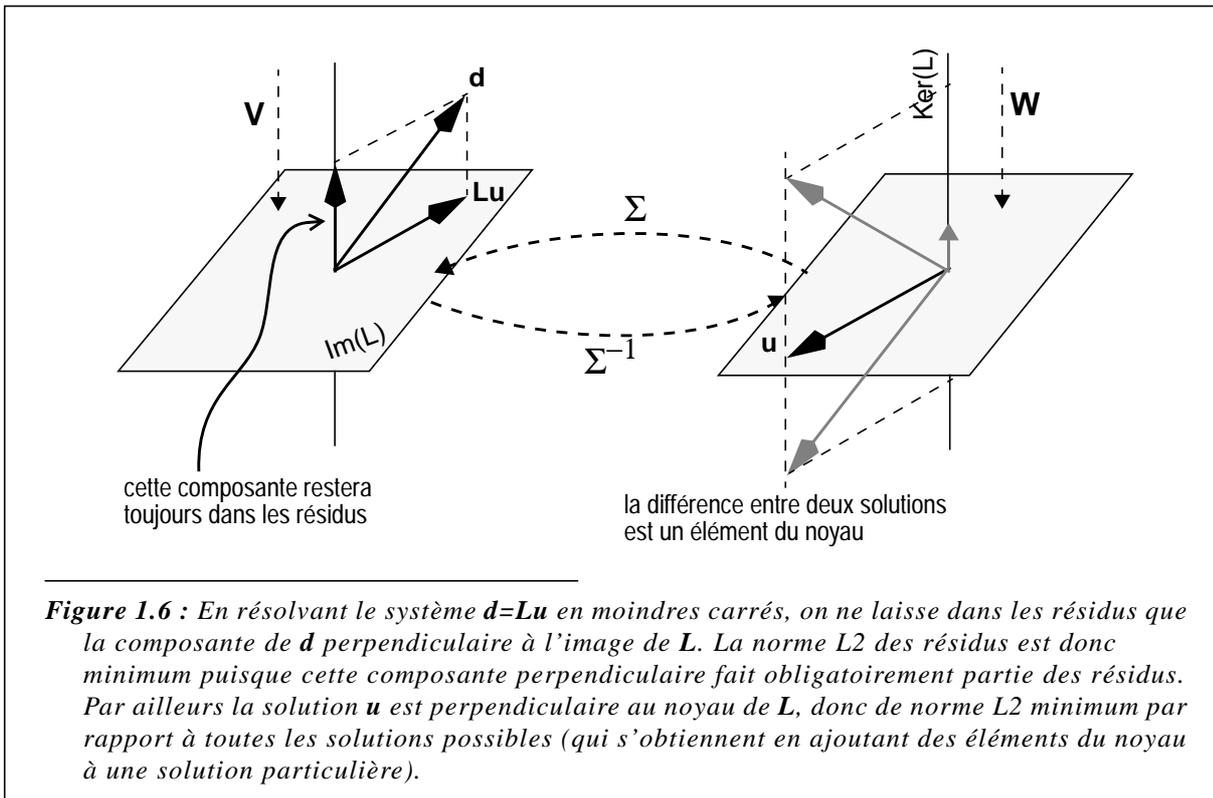
$$\boxed{\underline{\mathbf{L}} = \underline{\mathbf{V}}^* \underline{\Sigma} \underline{\mathbf{W}}} \quad (E1.32)$$

La matrice $\underline{\Sigma}$ est carrée diagonale et ne contient que les valeurs singulières non nulles de $\underline{\mathbf{L}}$ (dans la pratique on peut choisir quelles valeurs singulières on met dans cette matrice, souvent en fixant un seuil). $\underline{\mathbf{W}}$ et $\underline{\mathbf{V}}$ sont des projections, respectivement sur l'espace orthogonal au noyau de $\underline{\mathbf{L}}$ et sur l'image de $\underline{\mathbf{L}}$, et sont donc unitaires ($\underline{\mathbf{V}}\underline{\mathbf{V}}^* = \underline{\mathbf{I}}_d$ et $\underline{\mathbf{W}}\underline{\mathbf{W}}^* = \underline{\mathbf{I}}_d$). $\underline{\Sigma}$ établit une bijection entre ces deux espaces (voir la figure 1.6).

L'**inverse généralisé** de $\underline{\mathbf{L}}$ se contente d'inverser les valeurs singulières non nulles:

$$\boxed{\underline{\mathbf{L}}^\dagger = \underline{\mathbf{W}}^* \underline{\Sigma}^{-1} \underline{\mathbf{V}}} \quad (E1.33)$$

La solution ainsi obtenue d'une part explique toutes les données dans l'image de $\underline{\mathbf{L}}$ et d'autre part n'a pas de composante dans le noyau de $\underline{\mathbf{L}}$. La norme L2 de la solution est donc minimale (en ajoutant des éléments du noyau à une solution particulière, on obtient toutes les solutions possibles; la solution de norme L2 minimale est donc celle qui n'a pas de composante dans le noyau), ainsi que celle des résidus (les données hors de l'espace image se retrouvent dans tous les cas dans les résidus). D'où le nom de moindres carrés (voir fig. 1.6).



Il existe en fait deux formulations pour les inverses en moindres carrés (Kostov, 1989); à gauche (le plus classique, utilisé en général pour résoudre les systèmes sur-déterminés) et à droite (pour les systèmes sous-déterminés):

$$\boxed{\mathbf{u} = (\underline{\mathbf{L}}^* \underline{\mathbf{L}})^{-1} \underline{\mathbf{L}}^* \mathbf{d} \quad \mathbf{u} = \underline{\mathbf{L}}^* (\underline{\mathbf{L}} \underline{\mathbf{L}}^*)^{-1} \mathbf{d}} \quad (E1.34)$$

Les matrices $\underline{\mathbf{L}}^* \underline{\mathbf{L}}$ et $\underline{\mathbf{L}} \underline{\mathbf{L}}^*$ ne sont pas forcément inversibles. On peut alors effectuer leurs SVD d'après celle de $\underline{\mathbf{L}}$ afin d'obtenir leurs inverses généralisés:

$$\begin{aligned} \underline{\mathbf{L}}^* \underline{\mathbf{L}} &= \underline{\mathbf{W}}^* \underline{\Sigma}^2 \underline{\mathbf{W}} & \underline{\mathbf{L}} \underline{\mathbf{L}}^* &= \underline{\mathbf{V}}^* \underline{\Sigma}^2 \underline{\mathbf{V}} \\ (\underline{\mathbf{L}}^* \underline{\mathbf{L}})^\dagger &= \underline{\mathbf{W}}^* \underline{\Sigma}^{-2} \underline{\mathbf{W}} & (\underline{\mathbf{L}} \underline{\mathbf{L}}^*)^\dagger &= \underline{\mathbf{V}}^* \underline{\Sigma}^{-2} \underline{\mathbf{V}} \\ (\underline{\mathbf{L}}^* \underline{\mathbf{L}})^\dagger \underline{\mathbf{L}}^* &= \underline{\mathbf{W}}^* \underline{\Sigma}^{-1} \underline{\mathbf{V}} & \underline{\mathbf{L}}^* (\underline{\mathbf{L}} \underline{\mathbf{L}}^*)^\dagger &= \underline{\mathbf{W}}^* \underline{\Sigma}^{-1} \underline{\mathbf{V}} \end{aligned}$$

Ces deux inverses généralisés sont également égaux à l'inverse généralisé de $\underline{\mathbf{L}}$. L'inverse à gauche est utilisé préférentiellement lorsque le système est sur-déterminé (plus d'équations que d'inconnues), l'inverse à droite lorsque le système est sous-déterminé (plus d'inconnues que d'équations).

On peut également ajouter une petite valeur le long de la diagonale afin d'assurer l'invertibilité:

$$\boxed{\mathbf{u} = (\underline{\mathbf{L}}^* \underline{\mathbf{L}} + \varepsilon \mathbf{I})^{-1} \underline{\mathbf{L}}^* \mathbf{d} \quad \mathbf{u} = \underline{\mathbf{L}}^* (\underline{\mathbf{L}} \underline{\mathbf{L}}^* + \varepsilon \mathbf{I})^{-1} \mathbf{d}} \quad (E1.35)$$

ε est souvent appelé le **facteur d'amortissement (damping factor)**, et ces équations qualifiées de **moindres carrés amortis (damped least-squares)**. La présence du facteur d'amortissement est par ailleurs nécessaire dès lors que les données sont bruitées (voir chapitre 3).

La première expression (inverse à gauche) correspond également au minimum de la forme quadratique suivante, somme des carrés de la norme L2 des résidus et de la norme L2 pondérée du modèle:

$$\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}, \mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}) + \frac{\varepsilon}{2}(\mathbf{u}, \mathbf{u}) = \frac{1}{2}\|\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}\|^2 + \frac{\varepsilon}{2}\|\mathbf{u}\|^2 \quad (\text{E1.36})$$

Le minimum est en effet atteint quand le gradient s'annule:

$$\begin{aligned} \varphi(\mathbf{u}) &= \frac{1}{2}(\mathbf{d}, \mathbf{d}) - (\mathbf{d}, \tilde{\mathbf{L}}\mathbf{u}) + \frac{1}{2}(\tilde{\mathbf{L}}\mathbf{u}, \tilde{\mathbf{L}}\mathbf{u}) + \frac{\varepsilon}{2}(\mathbf{u}, \mathbf{u}) \\ &= \frac{1}{2}(\mathbf{d}, \mathbf{d}) - (\tilde{\mathbf{L}}^*\mathbf{d}, \mathbf{u}) + \frac{1}{2}((\tilde{\mathbf{L}}^*\tilde{\mathbf{L}} + \varepsilon\mathbf{I})\mathbf{u}, \mathbf{u}) \end{aligned}$$

$$\nabla\varphi(\mathbf{u}) = -\tilde{\mathbf{L}}^*\mathbf{d} + (\tilde{\mathbf{L}}^*\tilde{\mathbf{L}} + \varepsilon\mathbf{I})\mathbf{u}$$

D'où le résultat (E1.35).

1.7.2 - Commentaires

La décomposition en norme L2, par rapport à la simple transformée (paragr. 1.3.2), résout (évidemment) le problème de la non-inversibilité (y compris pour des données irrégulières), mais n'améliore que partiellement la focalisation (une droite ne se transforme toujours pas en point dans la transformée de Radon linéaire), n'atténue que partiellement les effets de bord, et ne résout pas du tout les problèmes d'aliasing.

Il existe d'autres critères que les moindres carrés pour l'inversion, mais ceux-ci possèdent l'avantage d'être bien connus mathématiquement et de donner lieu à des implémentations faciles et efficaces (d'un point de vue temps de calcul), d'où leur grand succès dans l'industrie.

1.7.3 - Illustrations

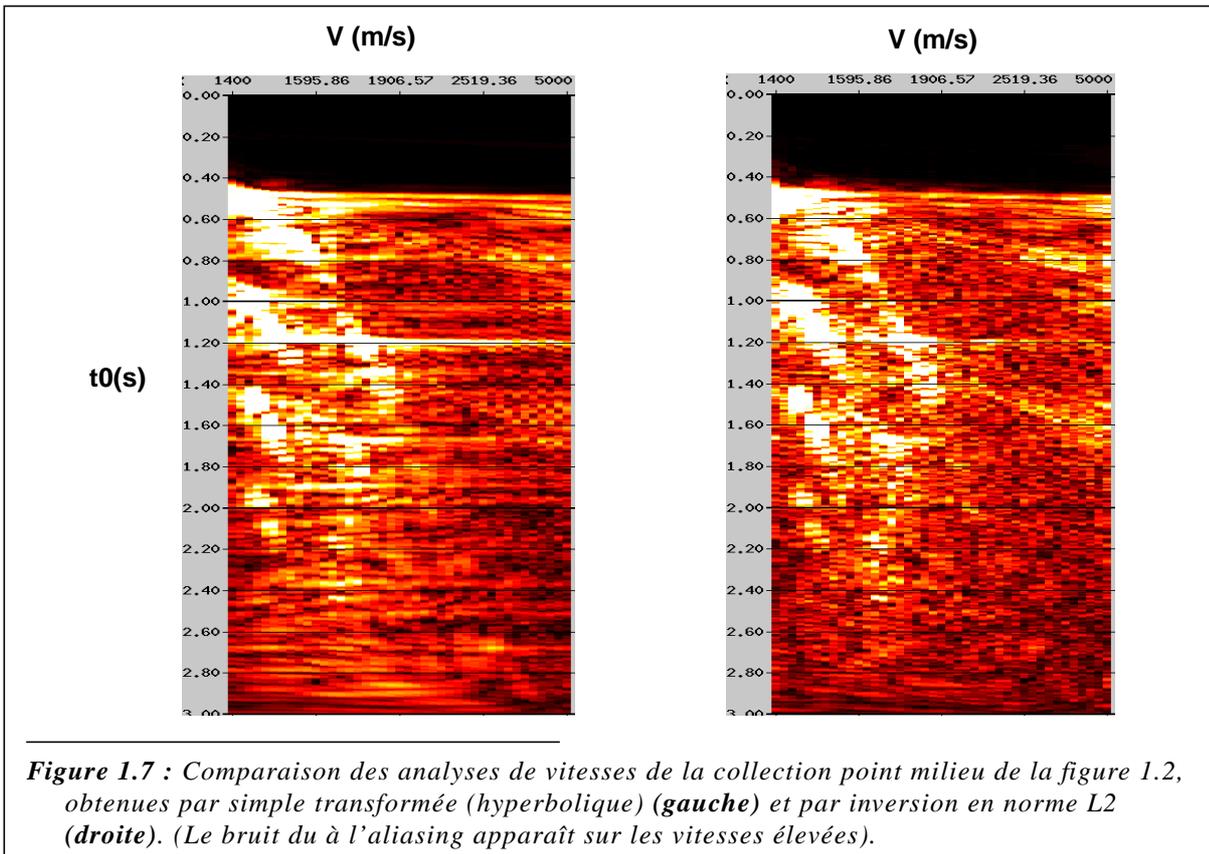
La figure 1.2 compare des analyses de vitesses obtenues par simple transformée de Radon hyperbolique (hyperboles V-SCAN) et par inversion en norme L2: l'inversion améliore la focalisation, réduit les traînées dues aux effets de bords, mais amplifie le bruit du à l'aliasing spatial. L'avantage principal de l'inversion reste toutefois la possibilité de reconstruire les données initiales (ce qui est impossible à partir de la transformée)

La figure 1.8 illustre, pour une collection point milieu (avec et sans corrections NMO), les trois transformées (opérateurs adjoints) et décompositions en norme L2 (E1.36): linéaire, parabolique, et hyperbolique (hyperboles P-SCAN).

1.8 - Discrétisation

La discrétisation des variables, et donc des équations, a été étudiée par Beylkin (1987). Il a montré que **les versions discrètes des équations doivent être considérées comme un problème à part entière**, au même titre que les versions continues, et en particulier que l'inverse de la transformée discrète n'était pas la version discrète de l'inverse de la transformée continue.

Les variables (t, x, t0, x0) sont discrétisées aux pas réguliers de (Δt , Δx , Δt_0 , Δx_0). On suppose ici que $\Delta t_0 = \Delta t$.



1.8.1 - Modélisation

Pour discrétiser l'opérateur de modélisation, on conserve le concept de superposition courbes de l'équ. (E1.21):

$$\underline{\mathbf{L}}\mathbf{u}(t, x) = \sum_{x_0} \sum_{t_0} \mathbf{u}(t_0, x_0) \psi(t - H(t_0, x_0, x)) \quad (\text{E1.37})$$

$$\psi(t) = \begin{cases} 1 - \frac{|t|}{\Delta t} & |t| \leq \Delta t \\ 0 & |t| \geq \Delta t \end{cases}$$

La fonction ψ remplace ici la fonction de dirac: l'élément $\mathbf{u}(t_0, x_0)$ ne contribue à $\mathbf{d}(t, x)$ que si la courbe correspondante passe à moins d'un pas d'échantillonnage Δt du point (t, x) (figure 1.9). La fonction ψ traduit l'opération d'éclatement (on éclate une valeur sur les deux échantillons les plus proches, d'une manière inversement proportionnelle à la distance aux échantillons). On peut envisager d'autres fonctions plus sophistiquées pour réaliser l'éclatement.

On aurait pu également discrétiser directement l'équation simplifiée (E1.22). Il faut dans ce cas calculer les valeurs de \mathbf{u} entre les échantillons:

$$\underline{\mathbf{L}}\mathbf{u}(t, x) = \sum_{x_0} \mathbf{u}_{\text{interp}}(G(t, x, x_0), x_0) G'_t(t, x, x_0) \quad (\text{E1.38})$$

$\mathbf{u}_{\text{interp}}(t_0, x_0)$ est la valeur de \mathbf{u} interpolée suivant t_0 . Une interpolation linéaire est la transposée de l'opération d'éclatement de l'équation (E1.37).

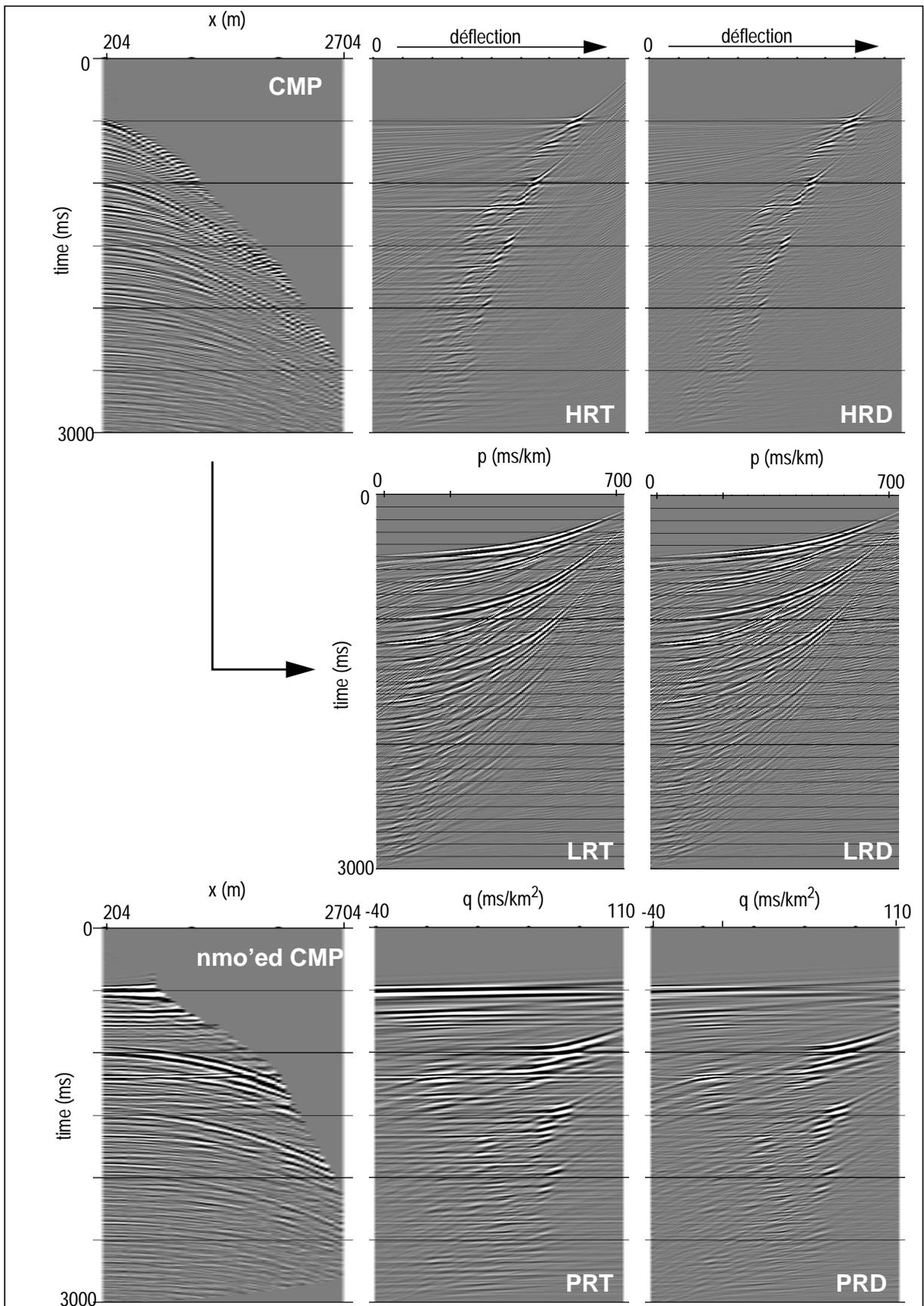


Figure 1.8 : Collection point milieu (à gauche) sans (haut) ou avec (milieu et bas) corrections NMO. Colonne du milieu, de haut en bas: Transformées de Radon Hyperbolique, Linéaire, et Parabolique. Colonne de droite, de haut en bas: les Décompositions de Radon (en norme L2). Famille des hyperboles P-SCAN: elles sont ici paramétrées par leurs déflexions à l'offset maximum.

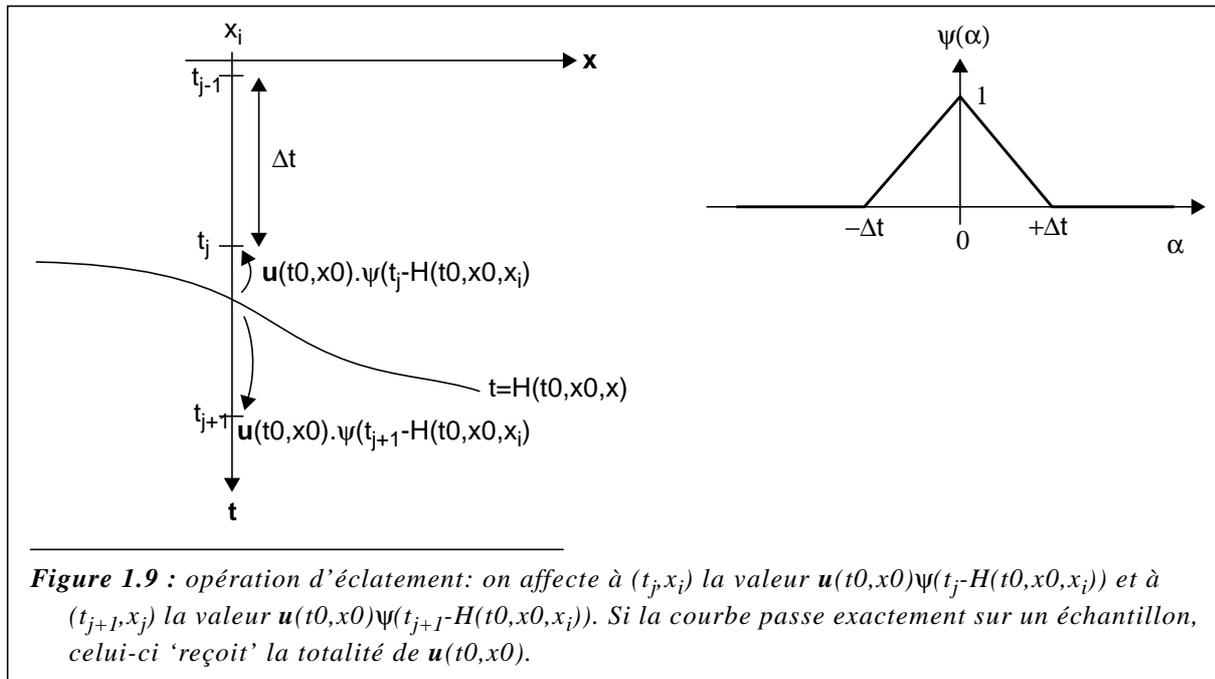


Figure 1.9 : opération d'éclatement: on affecte à (t_j, x_i) la valeur $u(t_0, x_0)\psi(t_j - H(t_0, x_0, x_i))$ et à (t_{j+1}, x_i) la valeur $u(t_0, x_0)\psi(t_{j+1} - H(t_0, x_0, x_i))$. Si la courbe passe exactement sur un échantillon, celui-ci 'reçoit' la totalité de $u(t_0, x_0)$.

Lemme: pour les familles de courbes invariantes par translation suivant t, les formules (E1.37) et (E1.38) sont identiques, et s'écrivent (voir la définition de h en (E1.3)):

$$\underline{L}u(t, x) = \sum_{x_0} u_{\text{interp}}(t - h(x_0, x), x_0) \tag{E1.39}$$

Démonstration:

Pour les familles invariantes par translation suivant t, (E1.37) s'écrit:

$$\underline{L}u(t, x) = \sum_{x_0} \sum_{t_0} u(t_0, x_0)\psi(t - (t_0 + h(x_0, x)))$$

Dans la seconde somme, la fonction ψ n'est nulle que pour les deux valeurs de t_0 qui encadrent la quantité $t - h(x_0, x)$. Soit $T_{\text{inf}}(t)$ la fonction qui donne la valeur discrétisée du temps immédiatement inférieure à t:

$$T_{\text{inf}}(t) = \max_k (k\Delta t, k \in \mathbb{N}, k\Delta t \leq t) \tag{E1.40}$$

Posons: $\tau(t, x_0, x) = T_{\text{inf}}(t - h(x_0, x))$
 $\delta\tau(t, x_0, x) = t - h(x_0, x) - \tau(t, x_0, x)$

La double somme s'écrit alors:

$$\underline{L}u(t, x) = \sum_{x_0} (u(\tau(t, x_0, x), x_0)\psi(\delta\tau(t, x_0, x)) + u(\tau(t, x_0, x) + \Delta t, x_0)\psi(\delta\tau(t, x_0, x) - \Delta t))$$

On vérifie facilement que $\psi(\delta\tau(t, x_0, x)) + \psi(\delta\tau(t, x_0, x) - \Delta t) = 1$, et que l'expression dans la somme est en fait une interpolation linéaire de u suivant t_0 . D'où le résultat.

Dans le cas général des familles de courbes non invariantes par translations suivant t, la discrétisation de (E1.21) n'est pas équivalente à la discrétisation de (E1.22). Dans l'optique du concept de superposition d'évènements d'amplitudes constantes, c'est la première expression qu'il faut retenir.

On a choisi ici de réaliser l'éclatement/interpolation linéairement: c'est une solution simple, mais qui ne conserve pas l'énergie. L'énergie d'une valeur éclatée sera par exemple différente suivant qu'elle tombe juste sur un échantillon (valeur 1 -> énergie $1^2=1$), ou juste entre deux échantillons (valeur 1

éclatée en deux fois 0.5 -> énergie $2*0.5^2=0.5$). Par ailleurs cette interpolation est très mauvaise pour des fréquences proches de Nyquist, ce qui oblige à travailler avec un sur-échantillonnage temporel.

D'autres interpolations sont envisageables (par splines, sinus cardinaux,...) respectant mieux les données (seule l'interpolation par sinus cardinaux conserve l'énergie, étant équivalente à une interpolation par transformée de Fourier). Mais celles-ci compliqueraient notablement les implémentations par rapport à l'interpolation linéaire. Notons enfin que ce problème d'interpolation ne se pose pas si l'on travaille dans le domaine de Fourier (les interpolations sont implicites par des sinus cardinaux).

1.8.2 - Opérateur adjoint de la modélisation

Pour calculer l'opérateur adjoint discrétisé, il faut d'abord définir des produits scalaires sur les domaines discrétisés. On choisit les produits scalaires euclidiens:

$$\begin{aligned} (\mathbf{d1}, \mathbf{d2})_{ID} &= \sum_x \sum_t \mathbf{d}_1(t, x) \mathbf{d}_2(t, x) \\ (\mathbf{u1}, \mathbf{u2})_{IU} &= \sum_{x0} \sum_{t0} \mathbf{u}_1(t0, x0) \mathbf{u}_2(t0, x0) \end{aligned} \tag{E1.41}$$

Théorème:

L'expression exacte de l'opérateur adjoint pour le problème discret est $\mathbf{d}_{interp}(t,x)$ est la valeur obtenue par interpolation linéaire suivant t):

$$\mathbf{L}^* \mathbf{d}(t0, x0) = \sum_x \mathbf{d}_{interp}(H(t0, x0, x), x) \tag{E1.42}$$

Démonstration:

Calculons la quantité:

$$\begin{aligned} (\mathbf{d}, \mathbf{L}\mathbf{u}) &= \sum_x \sum_t \mathbf{d}(t, x) \mathbf{L}\mathbf{u}(t, x) \\ &= \sum_x \sum_t \mathbf{d}(t, x) \sum_{x0} \sum_{t0} \mathbf{u}(t0, x0) \psi(t - H(t0, x0, x)) \\ &= \sum_{x0} \sum_{t0} \mathbf{u}(t0, x0) \sum_x \sum_t \mathbf{d}(t, x) \psi(t - H(t0, x0, x)) \end{aligned}$$

Dans la somme la plus interne, il ne peut y avoir que 2 valeurs de t pour lesquelles $\psi(t - H(t0, x0, x))$ est non nulle, celles qui encadrent la quantité $H(t0, x0, x)$.

Posons: $\tau(t0, x0, x) = T_{inf}(H(t0, x0, x))$

$\delta\tau(t0, x0, x) = H(t0, x0, x) - \tau(t, x0, x)$

On a alors, pour $x0, t0$, et x fixés:

$$\begin{aligned} &\sum_t \mathbf{d}(t, x) \psi(t - H(t0, x0, x)) \\ &= (\mathbf{d}(\tau(t0, x0, x), x) \psi(\delta\tau(t0, x0, x)) + \mathbf{d}(\tau(t0, x0, x) + \Delta t, x) \psi(\delta\tau(t0, x0, x) - \Delta t)) \\ &= \mathbf{d}_{interp}(H(t0, x0, x), x) \end{aligned}$$

Finalement:

$$(\mathbf{d}, \mathbf{L}\mathbf{u}) = \sum_{x0} \sum_{t0} \mathbf{u}(t0, x0) \sum_x \mathbf{d}_{interp}(H(t0, x0, x), x)$$

En identifiant cette expression à $(\underline{\mathbf{L}}^* \mathbf{d}, \mathbf{u})$, on en déduit donc l'adjoint pour le problème discret.

A posteriori, on constate que cette expression est équivalente à celle que l'on aurait obtenue en discrétisant l'expression de l'adjoint du problème continu (équ. (E1.30)).

1.9 - Expressions dans le domaine de Fourier

1.9.1 - Expressions de la transformée et de la modélisation de Radon

Trorey (1961) est le premier à avoir utilisé le domaine de Fourier pour la transformée de Radon linéaire, afin de montrer qu'elle n'occasionnait pas de perte d'information. Beylkin (1987) a développé la méthode pour calculer l'inverse exact de la transformée de Radon discrète.

On se restreint ici aux familles de courbes invariantes par translation suivant t (droites, paraboles, hyperboles P-SCAN), ce qui entraîne les propriétés suivantes.

$$\begin{aligned} t &= H(t_0, x_0, x) = t_0 + h(x_0, x) \\ t_0 &= G(x, t, x_0) = t - h(x_0, x) \\ G'_t(x, t, x_0) &= H'_{t_0}(t_0, x_0, x) = 1 \end{aligned} \tag{E1.43}$$

Théorème:

Dans le domaine de Fourier, où ω est la pulsation temporelle, l'opérateur de modélisation et son adjoint s'écrivent²:

$$\begin{aligned} \underline{\hat{\mathbf{L}}}\hat{\mathbf{u}}(\omega, x) &= \int_{x_0} \hat{\mathbf{u}}(\omega, x_0) e^{-i\omega h(x_0, x)} dx_0 \\ \underline{\hat{\mathbf{L}}}^* \hat{\mathbf{d}}(\omega, x_0) &= \int_x \hat{\mathbf{d}}(\omega, x) e^{i\omega h(x_0, x)} dx \end{aligned} \tag{E1.44}$$

Démonstration:

Reprenons l'expression de $\underline{\mathbf{L}}$ (E1.22):

$$\begin{aligned} \underline{\mathbf{L}}\mathbf{u}(t, x) &= \int_{x_0} \mathbf{u}(G(x_0, x, t), x_0) G'_t(x_0, x, t) dx_0 \\ &= \int_{x_0} \mathbf{u}(G(x_0, x, t), x_0) dx_0 \end{aligned}$$

Appliquons une transformée de Fourier suivant t à cette équation:

$$\begin{aligned} \underline{\hat{\mathbf{L}}}\hat{\mathbf{u}}(\omega, x) &= (\underline{\hat{\mathbf{L}}}\mathbf{u})(\omega, x) = \int_t \underline{\mathbf{L}}\mathbf{u}(t, x) e^{-i\omega t} dt \\ &= \int_t \int_{x_0} \mathbf{u}(G(x_0, x, t), x_0) e^{-i\omega t} dx_0 dt \end{aligned}$$

² Par convention on utilise le chapeau (^) pour désigner la Transformée de Fourier suivant la variable temps.

On applique le changement de variable $t_0=t-h(x_0,x)$:

$$\begin{aligned}\hat{\mathbf{L}}\hat{\mathbf{u}}(\omega, x) &= \int_{x_0} \int_t \mathbf{u}(t-h(x_0, x), x_0) e^{-i\omega t} dt dx_0 \\ &= \int_{x_0} \int_{t_0} \mathbf{u}(t_0, x_0) e^{-i\omega(t_0+h(x_0, x))} dt_0 dx_0 \\ &= \int_{x_0} \int_{t_0} \mathbf{u}(t_0, x_0) e^{-i\omega t_0} dt_0 e^{-i\omega h(x_0, x)} dx_0\end{aligned}$$

On reconnaît dans l'intégrale la transformée de Fourier de \mathbf{u} :

$$\hat{\mathbf{L}}\hat{\mathbf{u}}(\omega, x) = \int_{x_0} \hat{\mathbf{u}}(\omega, x_0) e^{-i\omega h(x_0, x)} dx_0$$

On procède exactement de la même façon pour l'opérateur adjoint.

La propriété d'invariance par translation suivant t se traduit par un **découplage entre les fréquences**: pour calculer les valeurs de $\hat{\mathbf{L}}\hat{\mathbf{u}}$ à une fréquence donnée, on n'utilise pas les valeurs de $\hat{\mathbf{u}}$ provenant d'autres fréquences.

Théorème: Pour les familles de courbes où la variable x_0 peut être mise en facteur ($t = t_0 + h(x_0, x) = t_0 + x_0 \cdot m(x)$), l'opérateur composé $\mathbf{L}^*\mathbf{L}$ s'exprime dans le domaine de Fourier comme une convolution suivant x_0 :

$$\hat{\mathbf{L}}^* \hat{\mathbf{L}} \hat{\mathbf{u}}(\omega, x_0) = \hat{\mathbf{u}}(\omega, x_0) *_{x_0} \sigma(\omega, x_0) \quad (\text{E1.45})$$

$$\sigma(\omega, x_0) = \int_x e^{i\omega m(x)x_0} dx \quad (\text{E1.46})$$

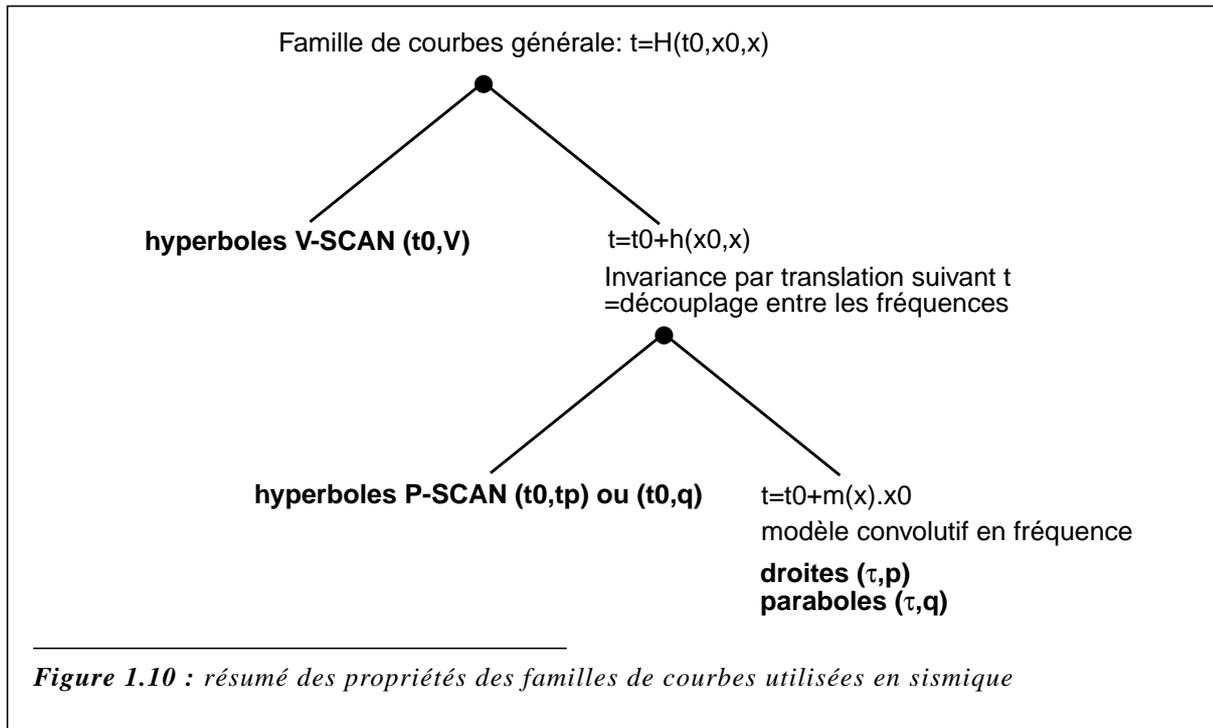
Démonstration:

En combinant les deux expressions de (E1.44):

$$\begin{aligned}\hat{\mathbf{L}}^* \hat{\mathbf{L}} \hat{\mathbf{u}}(\omega, x_0) &= \int_x \left(\int_{x'_0} \hat{\mathbf{u}}(\omega, x'_0) e^{-i\omega h(x'_0, x)} dx'_0 \right) e^{i\omega h(x_0, x)} dx \\ &= \int_{x'_0} \hat{\mathbf{u}}(\omega, x'_0) \int_x e^{i\omega(h(x_0, x) - h(x'_0, x))} dx dx'_0 \\ &= \int_{x'_0} \hat{\mathbf{u}}(\omega, x'_0) \int_x e^{i\omega(x_0 - x'_0)m(x)} dx dx'_0 \\ &= \int_{x'_0} \hat{\mathbf{u}}(\omega, x'_0) \sigma(\omega, x_0 - x'_0) dx'_0\end{aligned}$$

On reconnaît la convolution.

On peut obtenir cette écriture pour la famille de droite τ - p ($h(p, x) = px$), et la famille de paraboles τ - q ($h(q, x) = qx^2$), mais pas pour les familles d'hyperboles. La figure 1.10 résume les propriétés des transformées pour les différentes familles de courbes utilisées.



1.9.2 - Relation entre la Transformée de Radon Linéaire et la Transformée de Fourier 2D

Dans le domaine des fréquences temporelles, l'opérateur **R** pour la LRT s'écrit:

$$\begin{aligned} (\hat{\mathbf{R}}\hat{\mathbf{d}})(\omega, p) &= \int_x \hat{\mathbf{d}}(\omega, x) e^{i\omega p x} dx \\ &= \int_x \left(\int_t \mathbf{d}(t, x) e^{-i\omega t} dt \right) e^{i\omega p x} dx = \iint_{x,t} \mathbf{d}(t, x) e^{-i(\omega t - \omega p x)} dt dx \end{aligned}$$

Cette dernière expression est directement liée à la Transformée de Fourier 2D de **d**, qui s'écrit

$$(F_{(t,x)}\mathbf{d})(\omega, k_x) = \iint_{x,t} \mathbf{d}(t, x) e^{-i(\omega t + k_x x)} dt dx$$

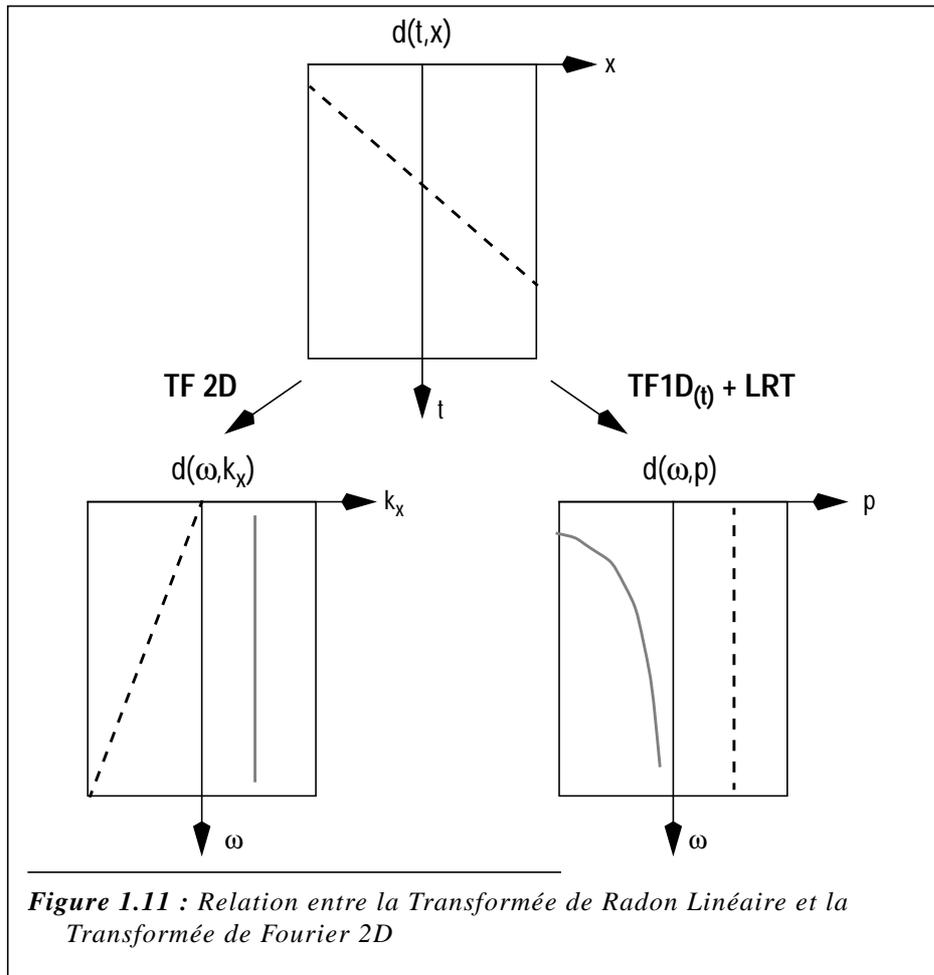
On constate en effet que:

$$\boxed{(\hat{\mathbf{R}}\hat{\mathbf{d}})(\omega, p) = (F_{(t,x)}\mathbf{d})(\omega, -\omega p)} \tag{E1.47}$$

Un axe $p=cste$ dans la LRT correspond notamment à une droite passant par l'origine dans la FT2D (fig. 1.11).

1.9.3 - Distributions sur D et U dans le domaine de Fourier

Nous avons vu (§ 1.3.4) qu'étendre les définitions des applications à des distributions n'était pas chose aisée. Le domaine de Fourier amène de ce point de vue quelques simplifications:



Famille des droites (τ, p)

Les équations définissant \mathbf{L} et \mathbf{R} pour des fonctions de L_2 sont, dans le domaine fréquentiel:

$$\begin{aligned} \hat{\mathbf{L}}\hat{\mathbf{u}}(\omega, x) &= \int_p \hat{\mathbf{u}}(\omega, p) e^{-i\omega p x} dp \\ \hat{\mathbf{R}}\hat{\mathbf{d}}(\omega, p) &= \int_x \hat{\mathbf{d}}(\omega, x) e^{i\omega p x} dx \end{aligned} \tag{E1.48}$$

Comme il y a un découplage entre les fréquences temporelles, on peut travailler sur chacune d'entre elles séparément. Pour une fréquence ω donnée:

Soit $\varphi, \psi \in S(\mathbb{R})$. Calculons les transformées de ces fonctions:

$$\begin{aligned} \hat{\mathbf{L}}_\omega \psi(x) &= \hat{\mathbf{L}}\psi(\omega, x) = \int_p \psi(p) e^{-i\omega p x} dp = \hat{\psi}(\omega x) \\ \hat{\mathbf{R}}_\omega \varphi(p) &= \hat{\mathbf{R}}\varphi(\omega, p) = \int_x \varphi(x) e^{i\omega p x} dx = \hat{\varphi}(-\omega p) \end{aligned} \tag{E1.49}$$

On sait que la transformée de Fourier d'une fonction de S est une fonction de S (Roddir, 1971). Donc, à condition que $\omega \neq 0$, les fonctions $\hat{\mathbf{L}}_\omega \psi$ et $\hat{\mathbf{R}}_\omega \varphi$ appartiennent également à S . La restriction à des

fréquences non nulles n'est pas vraiment un problème, les signaux sismiques étant en principe à moyenne nulle.

Cette fois on peut donc réellement définir les transformées de distributions tempérees:

$$\begin{aligned} \hat{\mathbf{d}}, \hat{\mathbf{u}} \in \mathcal{S}'(\mathbb{R}); \varphi, \psi \in \mathcal{S}(\mathbb{R}) \\ \langle \hat{\mathbf{L}}\hat{\mathbf{u}}, \varphi \rangle = \langle \hat{\mathbf{u}}, \hat{\mathbf{R}}\varphi \rangle \\ \langle \hat{\mathbf{R}}\hat{\mathbf{d}}, \psi \rangle = \langle \hat{\mathbf{d}}, \hat{\mathbf{L}}\psi \rangle \end{aligned} \tag{E1.50}$$

Transformée d'un dirac suivant p:

Lemme:

Pour la LRD, la modélisation d'un dirac 1D $\delta(p - P)$ suivant p est un évènement linéaire de pente P (s(t) est un signal temporel quelconque à moyenne nulle):

$$\hat{\mathbf{u}}(\omega, p) = \hat{\mathbf{s}}(\omega)\delta(p - P) \Rightarrow \underline{\mathbf{L}}\mathbf{u}(t, x) = \mathbf{s}(t + Px) \tag{E1.51}$$

Démonstration:

Calculons la modélisation de cette distribution:

$$\begin{aligned} \forall \varphi \in \mathcal{S} \\ \langle \hat{\mathbf{L}}\hat{\mathbf{u}}, \varphi \rangle &= \langle \hat{\mathbf{u}}, \hat{\mathbf{R}}\varphi \rangle = \hat{\mathbf{s}}(\omega) \langle \delta(p - P), \hat{\mathbf{R}}\varphi \rangle = \hat{\mathbf{s}}(\omega) \hat{\mathbf{R}}\varphi(P) \\ &= \hat{\mathbf{s}}(\omega) \int_x \varphi(x) e^{i\omega Px} dx \end{aligned}$$

On en déduit que:

$$\hat{\mathbf{L}}\hat{\mathbf{u}}(\omega, x) = \hat{\mathbf{s}}(\omega) e^{i\omega Px}$$

Puis on repasse dans le domaine temporel:

$$\underline{\mathbf{L}}\mathbf{u}(t, x) = \int_{\omega} \hat{\mathbf{L}}\hat{\mathbf{u}}(\omega, x) e^{i\omega t} d\omega = \int_{\omega} \hat{\mathbf{s}}(\omega) e^{i\omega Px} e^{i\omega t} d\omega = \int_{\omega} \hat{\mathbf{s}}(\omega) e^{i\omega(t + Px)} d\omega$$

finalement:

$$\underline{\mathbf{L}}\mathbf{u}(t, x) = \mathbf{s}(t + Px)$$

1.9.4 - LRD et PRD par les moindres carrés

Pour des courbes telles que les droites et les paraboles, l'opérateur $\mathbf{L}^*\mathbf{L}$ est une convolution suivant x_0 par un opérateur connu pour chaque fréquence temporelle ω (équ. (E1.45)). Résoudre le problème en norme L2 revient donc à appliquer \mathbf{L}^* à \mathbf{d} , puis $(\mathbf{L}^*\mathbf{L})^{-1}$ qui est une déconvolution suivant x_0 , par un opérateur connu σ , pour chaque fréquence ω (voir fig. 1.12)

Une implémentation basée sur ce principe est très efficace, car il existe des algorithmes performants (le plus connu étant celui de Levinson) pour résoudre les déconvolutions.

Si $\hat{\mathbf{u}}_0$ est un modèle contenant des diracs, c'est à dire des évènements 'purs', $\hat{\mathbf{u}}_s$, le stack des données, apparaît comme une version 'étalée' de $\hat{\mathbf{u}}_0$, dans laquelle les diracs sont convolués par un signal σ . Déconvoluer $\hat{\mathbf{u}}_s$ par σ , c'est donc essayer de retrouver $\hat{\mathbf{u}}_0$ en supprimant l'effet de σ . On ne

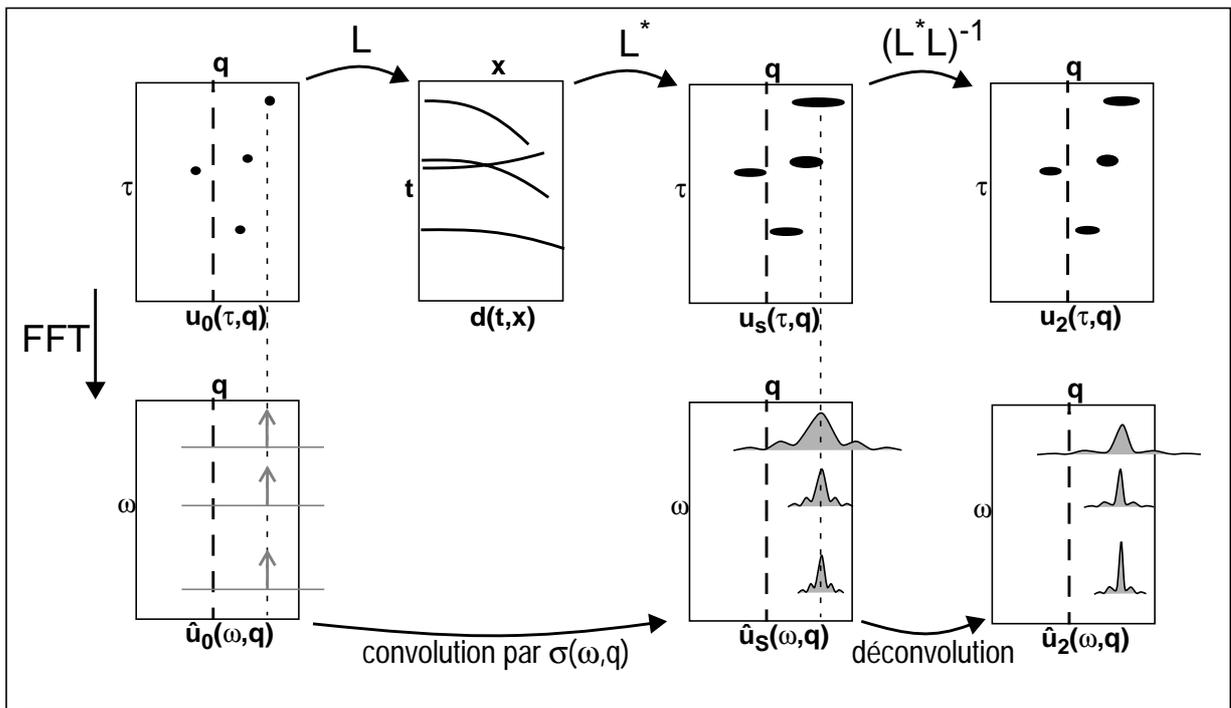


Figure 1.12 : Décomposition de Radon (ici Parabolique) en norme L2 dans le domaine de Fourier.

retrouvera toutefois pas exactement \hat{u}_0 , la convolution par σ ayant entraîné la perte de certaines informations.

1.9.5 - Opérateurs de convolution

1.9.5.1 - Variables continues

Les équations de la décomposition (en moindres carrés, paragr. 1.7) de Radon linéaire et parabolique peuvent s'écrire comme des convolutions/déconvolution suivant p ou q dans le domaine de Fourier. Il est donc intéressant d'étudier ces opérateurs, et principalement de connaître leurs spectres.

D'après l'équation (E1.46), les opérateurs de convolution s'écrivent, pour les deux transformées:

$$\begin{aligned} \sigma_{\text{lin}}(\omega, p) &= \int_{x_1}^{x_2} e^{i\omega p x} dx \\ \sigma_{\text{par}}(\omega, q) &= \int_{x_3}^{x_4} e^{i\omega q x^2} dx \end{aligned} \tag{E1.52}$$

Avec: $x_1 \leq x_2$ et $0 \leq x_3 \leq x_4$. Par ailleurs, nous ne considérons que les fréquences non nulles.

si $x_3 \leq x_4 \leq 0$, on a:

$$\sigma_{\text{par}}(\omega, q) = \int_{x_3}^{x_4} e^{i\omega q x^2} dx = -\int_{-x_3}^{-x_4} e^{i\omega q x^2} dx = \int_{-x_4}^{-x_3} e^{i\omega q x^2} dx$$

et on est ramené au cas précédent.

si $x_3 \leq 0 \leq x_4$, on a:

$$\sigma_{\text{par}}(\omega, q) = \int_{x_3}^0 e^{i\omega qx^2} dx + \int_0^{x_4} e^{i\omega qx^2} dx = \int_0^{-x_3} e^{i\omega qx^2} dx + \int_0^{x_4} e^{i\omega qx^2} dx$$

et on est à nouveau ramené au cas précédent.

Ces opérateurs agissant par convolution suivant p ou q , on calcule leurs spectres suivant ces variables. On note F la transformée de Fourier suivant $p|q$, et $k_{p|q}$ la variable de 'fréquence angulaire' associée à

$$p|q. \text{ Soit pour une fonction } s(p): \quad F_s(k_p) = \int_p s(p) \cdot e^{-ik_p p} \cdot dp$$

Théorème:

Les spectres des opérateurs de convolution (E1.52) s'écrivent:

$$\begin{aligned} F \sigma_{\text{lin}}(\omega, k_p) &= \begin{cases} \frac{2\pi}{|\omega|} & |\omega|x_1 \leq \text{sgn}(\omega)k_p \leq |\omega|x_2 \\ 0 & \text{sinon} \end{cases} \\ F \sigma_{\text{par}}(\omega, k_q) &= \begin{cases} \frac{\pi}{\sqrt{|\omega k_q|}} & |\omega|x_3^2 < \text{sgn}(\omega)k_q \leq |\omega|x_4^2 \\ 0 & \text{sinon} \end{cases} \end{aligned} \quad (\text{E1.53})$$

Démonstration:

Si l'on adopte le formalisme des distributions, on a, pour toute fonction $\varphi \in \mathcal{S}$:

$$\langle F_s, \varphi \rangle = \langle s, F\varphi \rangle$$

Commençons par l'opérateur σ_{lin} . On pose le changement de variable: $k_p = \omega x$:

$$\begin{aligned} \langle F \sigma_{\text{lin}}, \varphi \rangle &= \langle \sigma_{\text{lin}}, \hat{\varphi} \rangle \\ &= \int_p \int_{x_1}^{x_2} e^{i\omega p x} dx \hat{\varphi}(p) dp = \int_{x_1}^{x_2} \int_p \hat{\varphi}(p) e^{i\omega p x} dp dx \\ &= \int_{x_1}^{x_2} 2\pi \varphi(\omega x) dx = \int_{\omega x_1}^{\omega x_2} \frac{2\pi \varphi(k_p)}{\omega} dk_p \\ &= \text{sgn}(\omega) \int_{k_p} \frac{2\pi \omega x_1 \Pi_{\omega x_2}(k_p)}{\omega} \varphi(k_p) dk_p \end{aligned}$$

On en déduit que:

$$F \sigma_{\text{lin}}(\omega, k_p) = \omega x_1 \Pi_{\omega x_2}(k_p) \frac{2\pi}{|\omega|}$$

Pour le cas parabolique, le calcul est quasiment le même:

$$\begin{aligned} \langle F \sigma_{\text{par}}, \varphi \rangle &= \langle \sigma_{\text{lin}}, \hat{\varphi} \rangle \\ &= \int_q \int_{x_3}^{x_4} e^{i\omega q x^2} dx \hat{\varphi}(q) dq = \int_{x_3}^{x_4} \int_p \hat{\varphi}(q) e^{i\omega q x^2} dq dx \\ &= \int_{x_3}^{x_4} 2\pi \varphi(\omega x^2) dx \end{aligned}$$

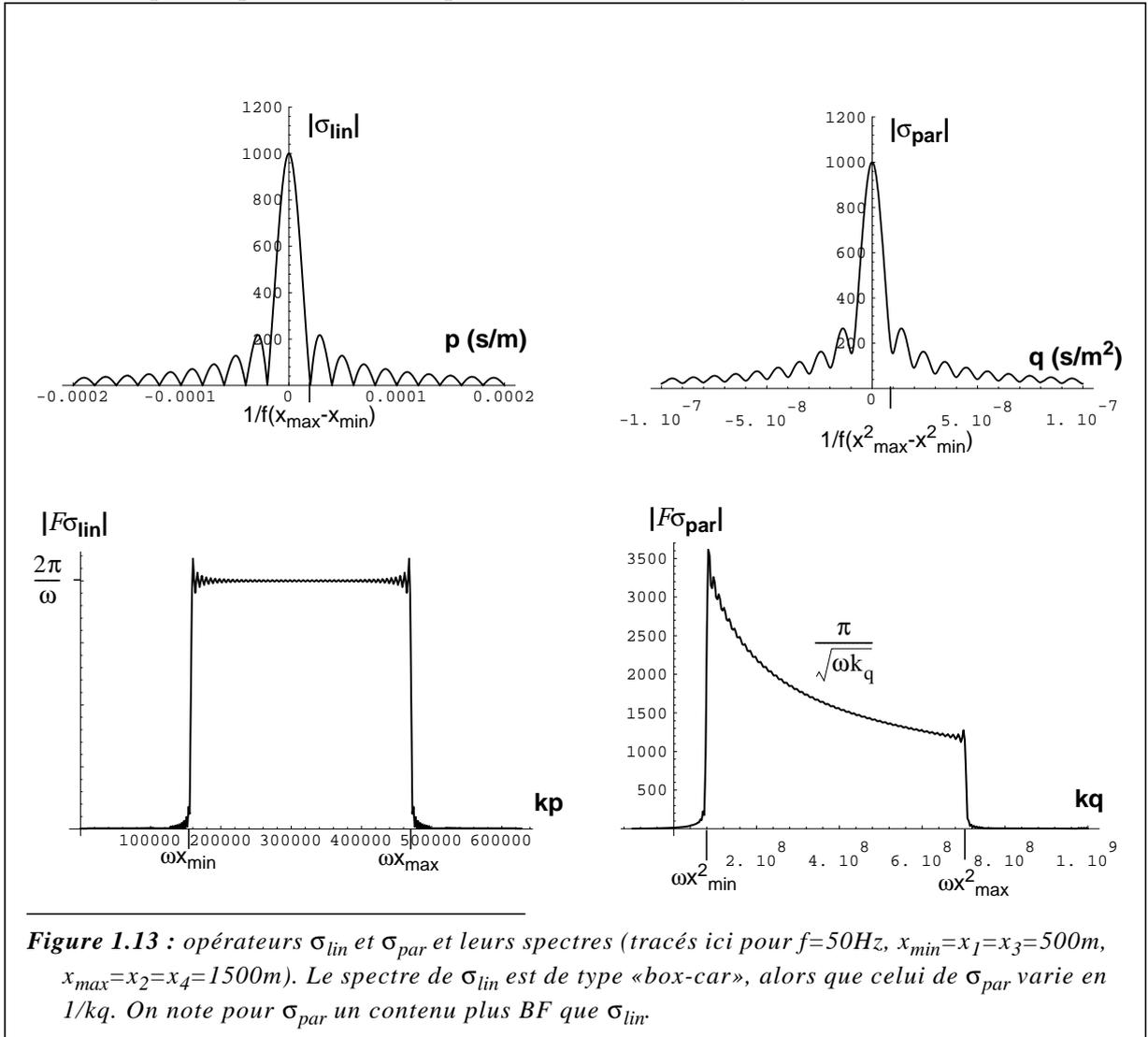
Posons le changement de variable $k_q = \omega x^2$, $dk_q = 2\omega x dx = 2 \text{sgn}(\omega) \sqrt{|\omega k_q|} dx$, ce qui donne:

$$\begin{aligned} \langle F \sigma_{par}, \varphi \rangle &= \text{sgn}(\omega) \int_{\omega x_3}^{\omega x_4} \frac{\pi \varphi(k_q)}{\sqrt{|\omega k_q|}} dk_q \\ &= \int_{k_q} \frac{\pi \omega x_3^2 \Pi_{\omega x_2^2}(k_q)}{\sqrt{|\omega k_q|}} \varphi(k_q) dk_q \end{aligned}$$

On en déduit que:

$$F \sigma_{par}(\omega, k_q) = \omega x_3^2 \Pi_{\omega x_2^2}(k_q) \frac{\pi}{\sqrt{|\omega k_q|}}$$

Des exemples d'opérateurs et leurs spectres sont tracés sur la fig. 1.13.



La principale différence entre les deux opérateurs réside dans leur spectre: celui de la LRD est constant, alors que celui de la PRD présente un contenu plus riche en BF. Outre leur action par rapport aux variables p et q, ils se comportent par ailleurs comme des filtres passe-bas pour les fréquences temporelles: en $1/|\omega|$ pour la LRT, et en $1/\sqrt{|\omega|}$ pour la PRT. C'est pourquoi l'application du filtre inverse -en $|\omega|$ pour la LRD, en $\sqrt{|\omega|}$ pour la PRD- permet de réaliser une approximation de l'inversion (cette approximation est meilleure pour la LRD que pour la PRD, car cette dernière nécessite en plus une déconvolution suivant q, même quand l'ouverture est infinie). Ce filtrage, décrit par Claerbout (1985) pour la LRD, et étendu à la PRD par Zhou and Greenhalgh (1994) est appelé **filtre rho (rho-filter)**.

Les transformées hyperboliques ne peuvent pas s'écrire avec des opérateurs de convolution. Mais par analogie, on peut essayer d'imaginer un filtre rho approximatif. En effet, vers l'origine $x=0$ les hyperboles sont approximables par des paraboles, et leur comportement asymptotique les rapprochent de droites. On peut donc prendre comme filtre rho une puissance de ω intermédiaire entre $\sqrt{|\omega|}$ et $|\omega|$, sans chercher de justification analytique.

Comportement pour des ouvertures infinies

On entend par ouvertures infinies le cas: $x_1 \rightarrow -\infty \quad x_3 = 0 \quad x_2, x_4 \rightarrow +\infty$

Le spectre de σ_{lin} tend vers la fonction constante $2\pi/|\omega|$, σ_{lin} tend donc vers un dirac centré à l'origine. Par contre, bien que le support du spectre de σ_{par} s'étende infiniment (vers les valeurs positives), σ_{par} ne tend pas vers un dirac. En effet:

$$\begin{aligned} \sigma_{par}(\omega, q) &= \int_0^{+\infty} e^{i\omega qx^2} dx = \int_0^{+\infty} (\cos(|\omega q|x^2) + i \operatorname{sgn}(\omega q) \sin(|\omega q|x^2)) dx \\ &= (1 + i \operatorname{sgn}(\omega q)) \sqrt{\frac{\pi}{8|\omega q|}} \end{aligned}$$

Pour des ouvertures infinies, on a donc:

$\begin{aligned} \sigma_{lin}(\omega, p) &= \frac{2\pi}{ \omega } \delta(p) \\ \sigma_{par}(\omega, q) &= (1 + i \operatorname{sgn}(\omega q)) \sqrt{\frac{\pi}{8 \omega q }} \end{aligned}$	(E1.54)
--	----------------

1.9.5.2 - variables discrètes

Les fréquences temporelles étant découplées, on ne s'intéresse pas ici à la discrétisation de t et τ . Les fonctions manipulées sont échantillonnées sur les valeurs de x et $p|q$ suivantes:

$$(x_1, x_2, \dots, x_{Nx}) \quad (p_1, p_2, \dots, p_{Np}) \quad (q_1, q_2, \dots, q_{Nq})$$

Comme on traite chaque fréquence indépendamment des autres, on omettra les indices correspondants.

Les données et les modèles sont des vecteurs de dimension Nx et Np :

$$\hat{\mathbf{d}} = \begin{bmatrix} d_1 \\ d_2 \\ \cdot \\ \cdot \\ d_{Nx} \end{bmatrix} \quad \hat{\mathbf{u}} = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ u_{Np} \end{bmatrix}$$

Les versions discrètes des opérations de modélisations s'écrivent $\hat{\mathbf{d}} = \hat{\mathbf{L}}\hat{\mathbf{u}}$, où les opérateurs \mathbf{L}_{lin} et \mathbf{L}_{par} sont des matrices de dimension $N_x.N_p$ et $N_x.N_q$:

$$\hat{\mathbf{L}}_{\text{lin}} = \begin{bmatrix} e^{-i\omega p_1 x_1} & e^{-i\omega p_2 x_1} & \dots & e^{-i\omega p_{N_p} x_1} \\ e^{-i\omega p_1 x_2} & e^{-i\omega p_2 x_2} & \dots & e^{-i\omega p_{N_p} x_2} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-i\omega p_1 x_{N_x}} & e^{-i\omega p_2 x_{N_x}} & \dots & e^{-i\omega p_{N_p} x_{N_x}} \end{bmatrix} \quad \hat{\mathbf{L}}_{\text{par}} = \begin{bmatrix} e^{-i\omega q_1 x_1^2} & e^{-i\omega q_2 x_1^2} & \dots & e^{-i\omega q_{N_q} x_1^2} \\ e^{-i\omega q_1 x_2^2} & e^{-i\omega q_2 x_2^2} & \dots & e^{-i\omega q_{N_q} x_2^2} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-i\omega q_1 x_{N_x}^2} & e^{-i\omega q_2 x_{N_x}^2} & \dots & e^{-i\omega q_{N_q} x_{N_x}^2} \end{bmatrix} \quad (\text{E1.55})$$

Les opérateurs adjoints \mathbf{L}^* sont les matrices hermitiennes (transposé+conjugué) de \mathbf{L} (et ce sont également les matrices des opérateurs de stack):

$$\hat{\mathbf{L}}_{\text{lin}}^* = \hat{\mathbf{R}}_{\text{lin}} = \hat{\mathbf{L}}_{\text{lin}}^H \quad \hat{\mathbf{L}}_{\text{par}}^* = \hat{\mathbf{R}}_{\text{par}} = \hat{\mathbf{L}}_{\text{par}}^H \quad (\text{E1.56})$$

D'où les matrices des opérateurs $\mathbf{L}^*\mathbf{L}$, carrées de dimensions N_p et N_q :

$$\boxed{[\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{lin}}]_{j,k} = \sum_{l=1}^{N_x} e^{i\omega(p_j - p_k)x_l} \quad [\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{par}}]_{j,k} = \sum_{l=1}^{N_x} e^{i\omega(q_j - q_k)x_l^2}} \quad (\text{E1.57})$$

Ces matrices sont hermitiennes. On peut noter également que chacun de leurs éléments est une approximation de l'intégrale (par la méthode des rectangles) définissant les opérateurs de convolution pour les variables continues (équ. (E1.53)):

$$[\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{lin}}]_{j,k} \approx \sigma_{\text{lin}}(\omega, p_j - p_k) \quad [\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{par}}]_{j,k} \approx \sigma_{\text{par}}(\omega, q_j - q_k) \quad (\text{E1.58})$$

Echantillonnage régulier en p et q

Si les variables p et q sont discrétisées à pas constant Δp et Δq :

$$p_j = p_{\text{min}} + (j-1)\Delta p \quad q_j = q_{\text{min}} + (j-1)\Delta q \quad (\text{E1.59})$$

Alors les opérateurs $\mathbf{L}^*\mathbf{L}$ ont une structure de **Toeplitz**. En effet:

$$[\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{lin}}]_{j,k} = \sum_{l=1}^{N_x} e^{i\omega(j-k)\Delta p x_l} \quad [\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{par}}]_{j,k} = \sum_{l=1}^{N_x} e^{i\omega(j-k)\Delta q x_l^2} \quad (\text{E1.60})$$

Les éléments, ne dépendant que de (k-j), sont constants le long de chaque diagonale:

$$\hat{\mathbf{L}}^* \hat{\mathbf{L}}_{\text{lin|par}} = \begin{bmatrix} \Sigma(\omega, 0) & \Sigma(\omega, 1)^* & \cdot & \cdot & \Sigma(\omega, Np-1)^* \\ \Sigma(\omega, 1) & \Sigma(\omega, 0) & & & \cdot \\ \cdot & & & & \cdot \\ \cdot & & & & \Sigma(\omega, 1)^* \\ \Sigma(\omega, Np-1) & \cdot & \cdot & \Sigma(\omega, 1) & \Sigma(\omega, 0) \end{bmatrix} \quad (E1.61)$$

$$\Sigma_{\text{lin}}(\omega, j) = \sum_{l=1}^{N_x} e^{i\omega j \Delta p x_l} \quad \Sigma_{\text{par}}(\omega, j) = \sum_{l=1}^{N_x} e^{i\omega j \Delta p x_l^2}$$

Il existe pour la résolution des systèmes de Toeplitz des algorithmes très efficaces, le plus connu étant celui de Levinson, qui nécessite de l'ordre de n^2 opérations au lieu des n^3 habituels pour un système quelconque. La résolution du problème inverse par les moindres carrés passant par la résolution des systèmes ci-dessus, on a donc tout intérêt à choisir un échantillonnage régulier en p et q.

Un échantillonnage régulier en p ou q entraîne une structure de Toeplitz pour les matrices $\mathbf{L}^* \mathbf{L}$. De même, un échantillonnage régulier en x entraîne une structure de Toeplitz pour la matrice $\mathbf{L} \mathbf{L}^*_{\text{lin}}$. Par contre, il faudrait un échantillonnage régulier en x^2 pour que la matrice $\mathbf{L} \mathbf{L}^*_{\text{par}}$ soit de Toeplitz. On ne retrouve pas de structure de Toeplitz dans les transformées hyperboliques (V-SCAN ou P-SCAN), ce qui limite l'intérêt de passer dans le domaine de Fourier pour ces transformées (dans le cas V-SCAN les fréquences ne sont même pas découplées).

1.10 - Paramétrisation

La paramétrisation est le choix d'une variable 'optimale' pour décrire l'espace des modèles. 'Optimal' est une considération subjective: cela peut vouloir dire 'ayant un sens physique', ou 'mathématiquement simple'. Ainsi dans la transformée τ -p, le paramètre p (la pente des droites) est-il naturel, ou bien en existe-t-il d'autres plus adaptés (l'angle avec l'horizontale par exemple)?

La réponse dépend de ce que l'on cherche à analyser: en sismique, on analyse des événements portant un signal temporel (l'ondelette sismique) que l'on retrouve à des temps différents pour des abscisses différentes. La paramétrisation par τ et p apparaît dans ce cas comme assez naturelle: on ne peut ainsi pas représenter un événement qui serait sur une abscisse constante, mais un tel événement ne correspond à rien de physique. Par ailleurs un échantillonnage régulier de la variable p échantillonne régulièrement le signal le long de l'axe du temps.

Une justification plus formelle est fournie par la formulation fréquentielle. Soit un modèle quelconque $\hat{\mathbf{u}}(\omega, p)$. Son image par l'opérateur composé $\hat{\mathbf{L}}^* \hat{\mathbf{L}}$ est obtenue par une convolution suivant p (équ. (E1.45)):

$$\hat{\mathbf{L}}^* \hat{\mathbf{L}} \hat{\mathbf{u}}(\omega, p) = \hat{\mathbf{u}}(\omega, p)^*_{\text{p}} \sigma(\omega, p)$$

Une translation du modèle sur l'axe p aboutit donc simplement à une translation de la même quantité de l'image de ce modèle:

$$\hat{\mathbf{u}}_p(\omega, p) = \hat{\mathbf{u}}(\omega, p - P)$$

$$\hat{\mathbf{L}}^* \hat{\mathbf{L}} \hat{\mathbf{u}}_p(\omega, p) = \hat{\mathbf{u}}_p(\omega, p)^*_{\text{p}} \sigma(\omega, p) = \hat{\mathbf{u}}(\omega, p - P)^*_{\text{p}} \sigma(\omega, p) = \hat{\mathbf{L}}^* \hat{\mathbf{L}} \hat{\mathbf{u}}(\omega, p - P)$$

L'image d'un évènement linéaire sur les données est donc la même quel que soit la pente p de cet évènement. Seule la position varie en fonction de la pente. Cela conduit à dire que la paramétrisation des droites par τ et p est 'naturelle' pour le problème posé.

On peut faire exactement le même raisonnement pour la transformée parabolique, pour dire que la paramétrisation des paraboles par τ et q est 'naturelle'.

Pour les hyperboles (V-SCAN et P-SCAN), le problème est plus complexe: on ne peut en effet pas exprimer l'opérateur composé $\hat{\mathbf{L}}^* \hat{\mathbf{L}}$ comme une convolution, et la forme de l'image d'un évènement hyperbolique dépend alors de la courbure de cet évènement.

Pour les hyperboles P-SCAN, on peut choisir une paramétrisation par analogie avec les paraboles. Si q est un paramètre naturel pour les paraboles, toute fonction linéaire de q est aussi un paramètre naturel. Or à une position d'abscisse X donnée, la déflexion des paraboles à cette valeur vaut X^2q , et la pente locale vaut $2Xq$ (fig. 1.14). On peut donc envisager de paramétrer les hyperboles P-SCAN par:

- (a) la courbure à l'origine
- (b) la déflexion à un offset donné
- (c) la pente à un offset donné.

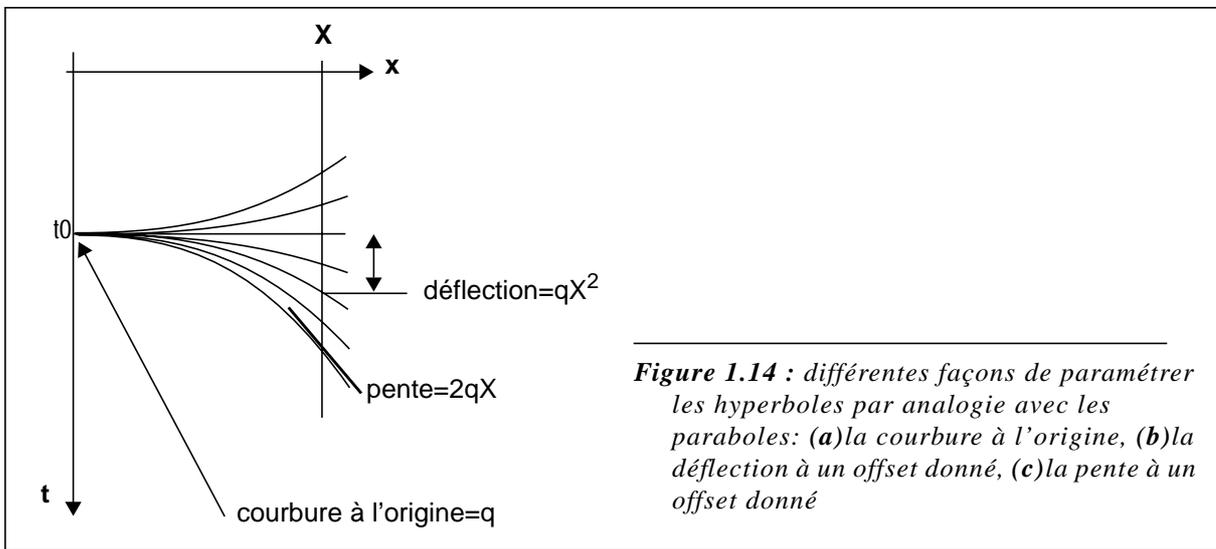


Figure 1.14 : différentes façons de paramétrer les hyperboles par analogie avec les paraboles: (a) la courbure à l'origine, (b) la déflexion à un offset donné, (c) la pente à un offset donné

La paramétrisation par la courbure q à l'origine équivaut, d'après (E1.8), à une paramétrisation par $1/t_p$. La paramétrisation la plus naturelle semble toutefois être la déflexion à un offset donné X , par exemple l'offset maximum des données. Si on note T_X cette déflexion, alors on a d'après (E1.7):

$$t_p(T_X) = \frac{X^2 - T_X^2 V_e^2}{2T_X V_e^2} \tag{E1.62}$$

1.11 - Conclusions

Nous avons défini et étudié les transformées de base pour le problème des décompositions de Radon, à savoir la modélisation et son adjoint (qui n'est autre que la transformée de Radon couramment utilisée), pour le problème continu et pour le problème discret, dans le domaine temps et dans le domaine de Fourier. La connaissance des opérateurs adjoints est utile aussi bien du point de vue théorique que du point de vue pratique (les implémentations de l'inversion par gradient conjugué utilisent les adjoints).

Dans le domaine de Fourier les équations se simplifient pour les familles de courbes invariantes par translation suivant le temps, et s'écrivent comme des convolutions pour la LRD et la PRD (convolutions suivant p et q respectivement). Dans ces derniers cas le problème inverse peut donc être résolu par une simple déconvolution. L'expression simple des opérateurs de convolution pour le problème continu à ouverture infinie permet de définir un 'filtre rho' très simple réalisant une bonne approximation de l'inverse exact du problème discret à ouverture limitée.

2 - Echantillonnage et Aliasing

2.1 - Introduction

L'aliasing est un problème à considérer dès lors que l'on discrétise des variables continues, ce qui est généralement imposé par les traitements numériques. La question est de déterminer d'une part les pas d'échantillonnage optimaux, et d'autre part les conséquences d'un sur- ou d'un sous-échantillonnage.

Nous nous intéressons principalement ici à la Décomposition de Radon Parabolique (PRD), parallèlement à la LRD, résolues en moindres carrés. Quatre variables interviennent: dans l'espace des données le temps t et l'offset x , et dans l'espace des modèles le temps d'intercept τ et la pente p pour la LRD, ou la courbure q pour la PRD¹.

Les problèmes d'aliasing suivant x et $p|q$ sont à considérer suivant deux points de vue différents: en général, à l'étape du traitement des données, celles-ci ont été acquises avec un certain échantillonnage spatial sur lequel il n'est plus possible de revenir, alors que l'on peut encore choisir un pas en $p|q$.

Concernant la LRD, l'échantillonnage en x et en p a été discuté par plusieurs auteurs (Stoffa et al., 1981; Maroof, 1984; Kostov, 1989), en s'appuyant sur le nombre d'onde $k=\omega p$. Turner (1990), à partir de raisonnements graphiques simples, a déterminé des relations liant le pas en offset Δx , la gamme de pentes étudiée ($p_{\max}-p_{\min}$), le pas entre deux pentes Δp , et l'ouverture en offset ($x_{\max}-x_{\min}$), en montrant par ailleurs les conséquences de l'aliasing.

En considérant la PRD comme une LRD après un changement de variable $X=x^2$, Kabir (1992) extrapole formellement les résultats précédents. La justification de cette extrapolation mérite d'être étudiée plus en détail: il n'est en effet peut-être pas loisible d'appliquer directement les relations de la LRD, établies sur la base de pas constants aussi bien en x qu'en p . Par ailleurs on souhaite mettre en évidence les différences de comportement entre LRD et PRD.

Pour l'étude des problèmes d'échantillonnage et d'aliasing, il est utile de comparer les deux transformations, linéaire et parabolique. Par ailleurs, le formalisme consistant à poser le problème comme une convolution/déconvolution suivant $p|q$ dans le domaine de Fourier $(\omega, p|q)$ (Zhou and Greenhalgh, 1994) est bien adapté car il ramène le problème à l'étude d'opérateurs de convolution.

2.2 - Echantillonnage temporel

Dans le cas de familles de courbes invariantes par translation suivant le temps, les bandes passantes des données et du modèle sont identiques, puisqu'il y a découplage entre les fréquences dans l'opérateur de modélisation (équ. (E1.44)). Dans ce cas les pas d'échantillonnages en temps Δt et Δt_0 peuvent être choisis identiques d'après la fréquence maximum des données: $\Delta t = \Delta t_0 = \frac{1}{2f_{\max}}$.

Dans le cas général, le rapport entre les bandes passantes dépend du stretch factor défini en (E1.26). Reprenons l'expression de la modélisation (équ (E1.21)):

$$(\mathbf{L}\mathbf{u})(t, x) = \iint_{x_0 t_0} \mathbf{u}(t_0, x_0) \delta(t - H(t_0, x_0, x)) dt_0 dx_0$$

La transformée de Fourier de cette relation est:

¹. Dans le texte on notera $p|q$ lorsque l'on voudra faire référence à la fois à la LRD et la PRD.

$$\begin{aligned} (\hat{\mathbf{L}}\hat{\mathbf{u}})(\omega, \mathbf{x}) &= \int \int \int_t \mathbf{u}(t0, \mathbf{x}0) \delta(t - H(t0, \mathbf{x}0, \mathbf{x})) dt0 d\mathbf{x}0 e^{-i\omega t} dt \\ &= \int \int_{\mathbf{x}0 t0} \mathbf{u}(t0, \mathbf{x}0) e^{-i\omega H(t0, \mathbf{x}0, \mathbf{x})} dt0 d\mathbf{x}0 \end{aligned}$$

Supposons que le stretch factor $S = \frac{1}{H_{t0}(t0, \mathbf{x}0, \mathbf{x})} = G'_t(t, \mathbf{x}, \mathbf{x}0)$ ne dépende pas du temps. On peut

alors écrire: $G(t, \mathbf{x}, \mathbf{x}0) = G(0, \mathbf{x}, \mathbf{x}0) + t S(\mathbf{x}0, \mathbf{x})$
 $H(t0, \mathbf{x}0, \mathbf{x}) = H(0, \mathbf{x}0, \mathbf{x}) + t0/S(\mathbf{x}0, \mathbf{x})$

D'où:

$$\begin{aligned} (\hat{\mathbf{L}}\hat{\mathbf{u}})(\omega, \mathbf{x}) &= \int \int_{\mathbf{x}0 t0} \mathbf{u}(t0, \mathbf{x}0) e^{-i\omega \frac{t0}{S(\mathbf{x}0, \mathbf{x})}} e^{-i\omega H(0, \mathbf{x}0, \mathbf{x})} dt0 d\mathbf{x}0 \\ &= \int_{\mathbf{x}0} \left(\int_{t0} \mathbf{u}(t0, \mathbf{x}0) e^{-i\frac{\omega}{S(\mathbf{x}0, \mathbf{x})} t0} dt0 \right) e^{-i\omega H(0, \mathbf{x}0, \mathbf{x})} d\mathbf{x}0 \\ &= \int_{\mathbf{x}0} \hat{\mathbf{u}}\left(\frac{\omega}{S(\mathbf{x}0, \mathbf{x})}, \mathbf{x}0\right) e^{-i\omega H(0, \mathbf{x}0, \mathbf{x})} d\mathbf{x}0 \end{aligned}$$

Cette dernière montre que le stretch factor induit une dilatation de la bande de fréquence entre les données et le modèle. En particulier, pour un stretch factor inférieur à 1.0 la fréquence maximum des données pourra être plus élevée que la fréquence maximum du modèle.

On se permet d'étendre cette conclusion au cas où le stretch factor n'est pas constant. Si le stretch factor est borné ($S_{\min} \leq S \leq S_{\max}$), et si on connaît la fréquence maximum f_{\max} des données, la fréquence

maximum F_{\max} du modèle est majorée par: $F_{\max} \leq \frac{f_{\max}}{S_{\min}}$; inversement si on connaît F_{\max} , f_{\max} est

majorée par: $f_{\max} \leq \frac{F_{\max}}{S_{\max}}$.

En particulier, pour les hyperboles V-SCAN: $H(t0, V, \mathbf{x}) = \sqrt{t0^2 + \frac{\mathbf{x}^2}{V^2}}$ et $S = \sqrt{1 + \frac{\mathbf{x}^2}{t0^2 V^2}} \geq 1$, le modèle ne peut donc pas contenir de fréquences supérieures aux fréquences des données.

2.3 - Echantillonnage en p|q

2.3.1 - Ouverture p|q infinie

La LRD et la PRD en moindres carrés pouvant être posées comme un problème de convolution/déconvolution par un opérateur connu dans le domaine de Fourier, les fréquences (k_p ou k_q) coupées par l'opérateur lors de convolution ne pourront pas être récupérées par la déconvolution. Par conséquent, la bande passante des objets que l'on manipule est limitée par le spectre des opérateurs de convolution. Les pas d'échantillonnage optimums en p|q peuvent donc être calculés à partir de ces spectres.

2.3 -Echantillonnage en p | q

Pour bien représenter toute l'information contenue dans un signal complexe $s(t)$, dont la largeur de bande est B_s (en pulsations), il faut choisir une fréquence d'échantillonnage $\omega_e \geq B_s$, c'est à dire un pas

$\Delta t = \frac{2\pi}{\omega_e} \leq \frac{2\pi}{B_s}$. Il s'agit là d'une condition suffisante qui évite le repliement des spectres.

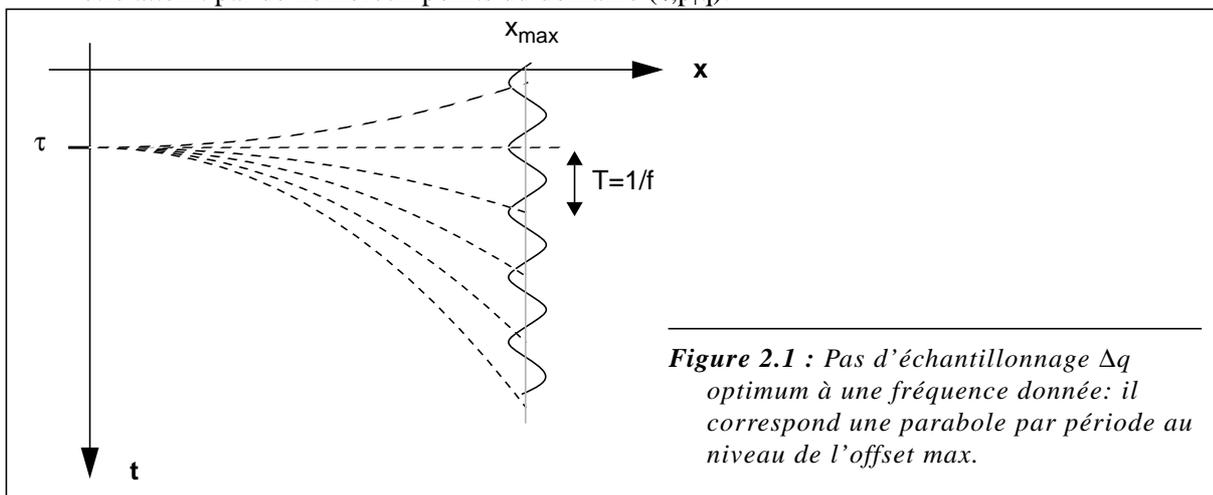
Pour la LRD, le spectre de σ_{lin} s'étend de $k_{pmin}=\omega x_{min}$ à $k_{pmax}=\omega x_{max}$ (équ. (E1.53) et fig. 1.13). Pour la PRD, le spectre est toujours nul pour $k_q < 0$; Il s'étend entre $k_{qmin}=\omega(x^2)_{min}$ et $k_{qmax}=\omega(x^2)_{max}$. Cela permet de définir les pas d'échantillonnage critiques:

$$\Delta p \leq \Delta p_c = \frac{1}{f(x_{max} - x_{min})}$$

$$\Delta q \leq \Delta q_c = \frac{1}{f((x^2)_{max} - (x^2)_{min})}$$
(E2.1)

...qui amènent plusieurs remarques:

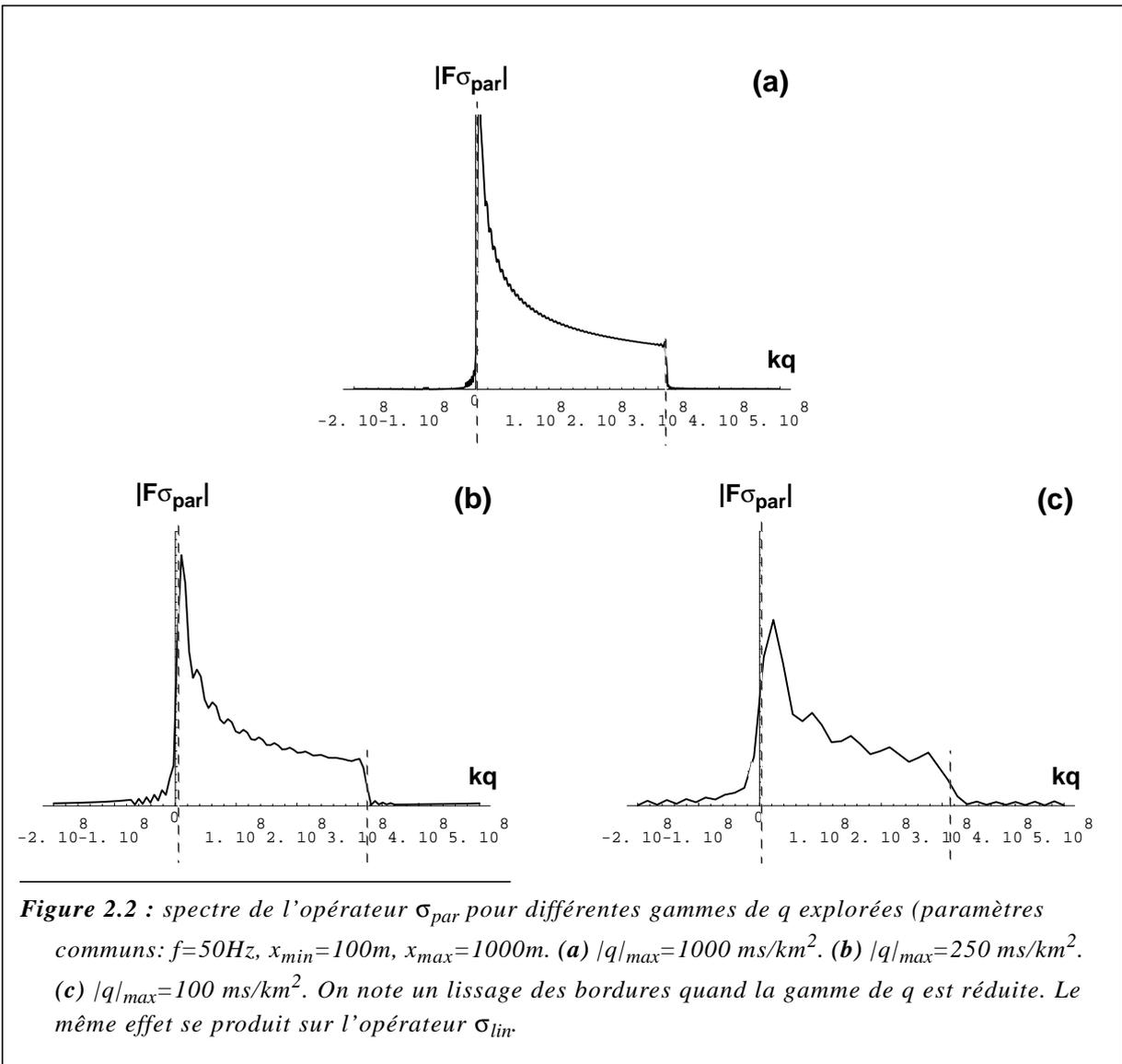
- Pour la PRD, dans les cas courants rencontrés en sismique (des valeurs courantes sont une gamme d'offsets de 100 à 3000m), le terme $(x^2)_{min}$ est souvent négligeable par rapport au terme $(x^2)_{max}$. Dans ce cas on obtient la même relation que Kabir (1992).
- La PRD apparaît à posteriori comme une extrapolation de la LRD, à ceci près qu'il n'y aurait que des offsets positifs.
- Le pas limite en p|q dépend bien sûr de la fréquence temporelle considérée.
- Si $x_{min}=0$, pour un temps d'intercept τ et une fréquence f donnés, le pas optimum correspond au niveau de l'offset maximum à un échantillonnage de 1 point par période (fig. 2.1). Un raisonnement simpliste sur la seule dimension temps de la trace à offset max aurait conduit à prendre au moins 2 points par période, soit un Δp deux fois plus petit. Or ici le problème est fondamentalement 2D, n'importe quel point du domaine (t,x) pouvant être atteint par de nombreux points du domaine $(\tau,p|q)$.



2.3.2 - Ouverture p | q finie

Les pas d'échantillonnage optimums ont été déterminés sur la base des spectres théoriques des opérateurs, c'est à dire en considérant que l'extension de p|q est infinie.

Lorsque la gamme des p|q utilisée est limitée, le spectre des opérateurs s'étalent (précisément, les spectres théoriques sont convolués par un sinus cardinal), et les coupures brutales sont d'autant plus lissées que la gamme de p|q est réduite (fig. 2.2).



La bande passante des opérateurs étant plus large, cela doit conduire à choisir un pas $\Delta p|q$ inférieur au pas optimal théorique.

L'élargissement du spectre est particulièrement sensible pour la PRD: le spectre croît en $1/\sqrt{k}$ au voisinage de 0, et peut donc atteindre de grandes valeurs et ainsi induire un effet de bord important si $(x^2)_{min}$ est faible devant $(x^2)_{max}$ (ce phénomène s'observe bien sur la figure 2.2, l'étalement du spectre est plus important sur la gauche, vers les fréquences négatives, que sur la droite).

2.3.3 - Conséquences d'un sur- ou d'un sous-échantillonnage en $p|q$

En théorie, un sur-échantillonnage en $p|q$ a les effets suivants:

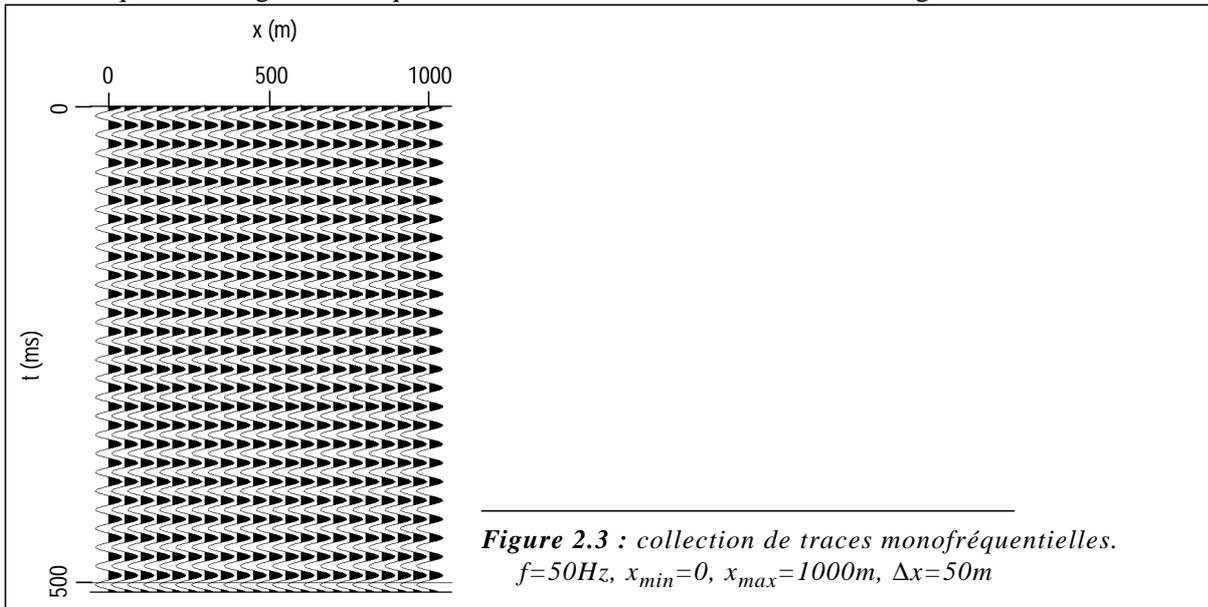
- du point de vue déconvolution, le spectre des opérateurs contient une plage nulle de plus en plus grande, obligeant ainsi à augmenter le facteur de pré-blanchiment.
- du point de vue inversion de matrice, la matrice $\mathbf{L}^* \mathbf{L}$ devient mal conditionnée (Kostov, 1990), ses colonnes étant quasi-identiques. Cela oblige donc à stabiliser plus fortement l'inversion par un facteur d'amortissement plus élevé.
- le temps de calcul augmente.

Kostov (1990) a également vérifié, pour la LRD, qu'un sur-échantillonnage ne permettait pas d'augmenter la focalisation des évènements.

Quel que soit le point de vue adopté, il ne semble donc pas souhaitable de sur-échantillonner en p|q. Il reste à vérifier que le cela n'apporterait pas des avantages susceptibles de compenser les inconvénients.

Quant au sous-échantillonnage, il induit un repliement du spectre des opérateurs, ce qui à priori devrait dégrader les résultats.

Pour vérifier ces effets en pratique, nous prenons une collection de traces monofréquentielle (f=50Hz) contenant un seul évènement horizontal (fig 2.3). Nous calculons la PRD de ces données avec plusieurs valeurs Δq, et nous regardons la qualité de la modélisation en mesurant l'énergie du résidu.



Le pas en q est défini comme une fraction du pas optimum: $\Delta q = \alpha \cdot \frac{1}{fx_{\max}^2}$.

Les PRD et leurs résidus pour différentes valeurs de α sont présentées fig. 2.4:

- Entre α=0.9 (échantillonnage limite) et α=0.3 (sur-échantillonnage), la qualité de la reconstruction s'améliore légèrement, sans plus. La focalisation de l'évènement n'est quant à elle pas améliorée.
- Pour α=0.9, le résidu est à peu près le même sur chaque trace. Par contre pour α=1.1 (léger sous-échantillonnage), les traces à offsets lointains ne sont plus du tout reconstruites, l'effet de seuil est évident. Les traces non reconstruites sont en fait celles sur lesquelles l'écart de temps entre 2 paraboles est supérieur à la période.

Ces 2 phénomènes se retrouvent sur la courbe de l'énergie résiduelle en fonction de α (fig. 2.5). Entre α=0.1 et α=0.9, la variation est faible, et l'effet de seuil se manifeste autour de α=1.

Sur la figure 2.6 nous prenons maintenant une collection de traces monofréquentielle ne contenant qu'une seule trace non nulle. On calcule la PRD de ces données avec plusieurs pas d'échantillonnage en q, puis on applique la modélisation. Avec un échantillonnage suffisant la trace est correctement reconstruite. En situation de sous-échantillonnage la trace n'est plus reconstruite correctement et donne naissance à un 'alias' situé à un offset qui dépend du facteur d'échantillonnage.

Ce phénomène est directement lié au repliement du spectre de l'opérateur σ_{par} (fig. 2.7): les informations concernant des offsets différents se trouvent mélangées, et par conséquent il devient impossible de différencier ces offsets.

La position des figures d'aliasing dépend de la position originelle de la trace non nulle et du facteur de sous-échantillonnage (fig. 2.8). Certaines traces ne s'aliasent pas (la trace 31 sur l'exemple fig. 2.8),

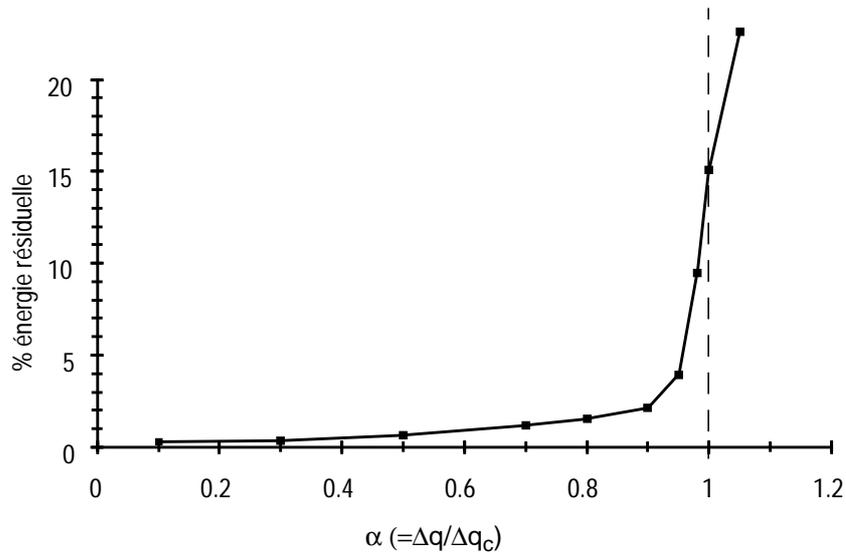


Figure 2.5 : Energie résiduelle (données fig. 2.3) suivant le pas d'échantillonnage en q

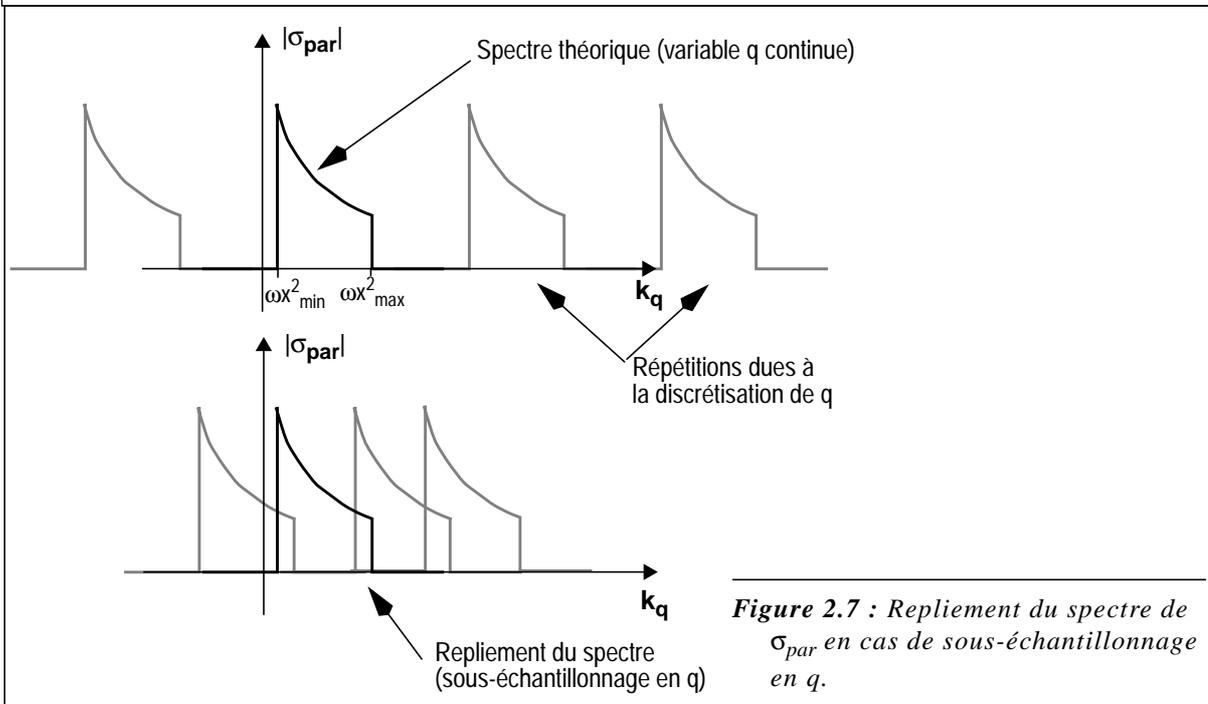


Figure 2.7 : Repliement du spectre de σ_{par} en cas de sous-échantillonnage en q .

car elles correspondent aux zones du spectre de σ_{par} non affectées par le repliement (fig. 2.7). Si le sous-échantillonnage devient trop sévère ($\alpha > 2$), de telles zones n'existent plus et toutes les traces s'aliasent.

Turner (1990) a présenté des exemples analogues pour la LRT. Il utilise une simple transformée et non une décomposition, ce qui change les amplitudes sur le résultat final mais les conclusions sont qualitativement les mêmes.

2.3.4 - Echantillonnage pour la HRT (hyperboles P-SCAN)

Le problème de la paramétrisation et de l'échantillonnage des hyperboles P-SCAN (et plus encore V-SCAN) est plus complexe que pour les droites et les paraboles, notamment à cause d'une écriture moins simple des équations dans le domaine de Fourier (voir chapitre 1).

A partir du même type de raisonnement graphique que Turner (1990), on peut néanmoins déterminer un pas d'échantillonnage minimum: par la modélisation chaque point de l'espace (t,x) est reconstruit par une intégrale le long d'une trajectoire dans le domaine (t_0, t_p) (équ. (E1.22)), ou (t_0, T_X) si on paramétrise

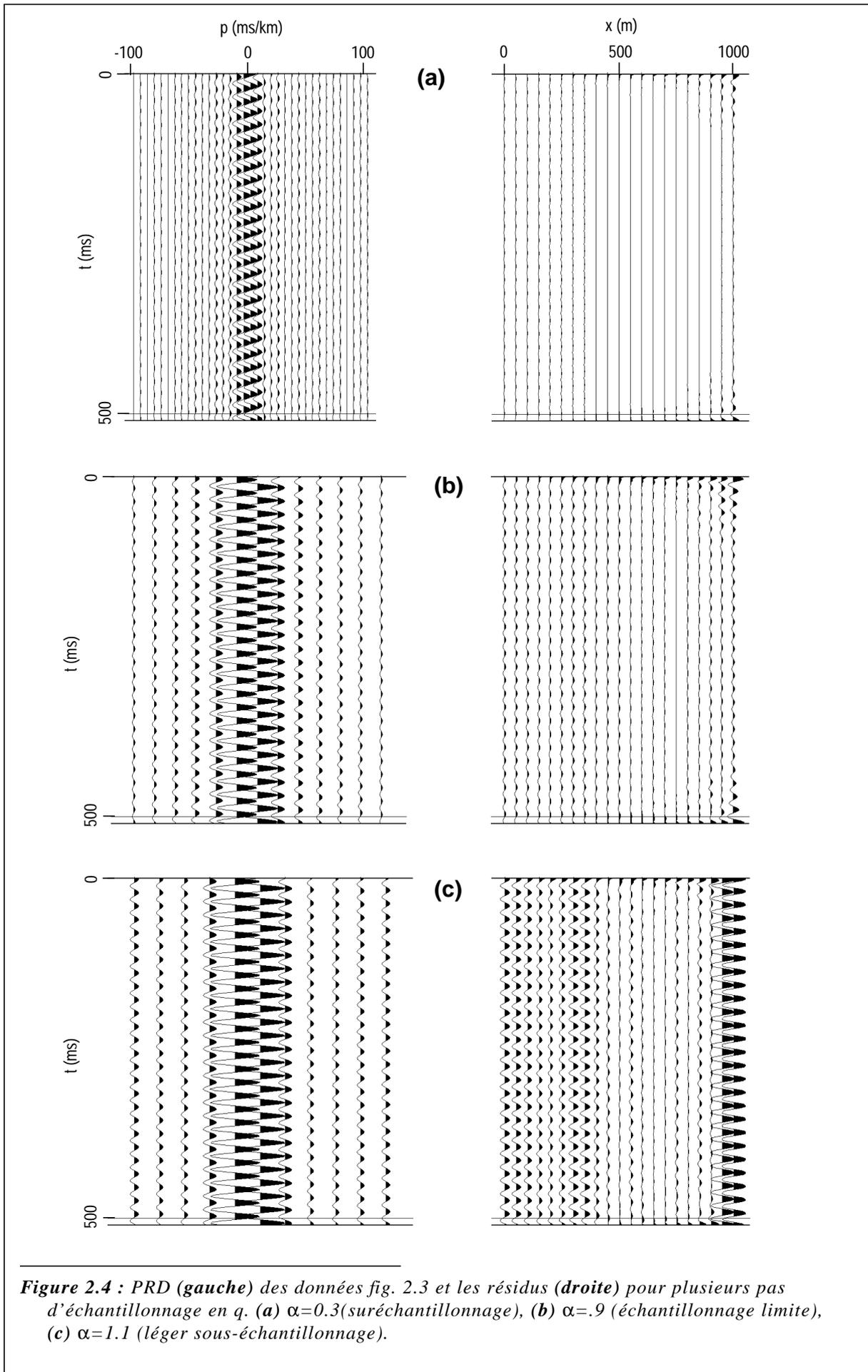


Figure 2.4 : PRD (gauche) des données fig. 2.3 et les résidus (droite) pour plusieurs pas d'échantillonnage en q . (a) $\alpha=0.3$ (suréchantillonnage), (b) $\alpha=.9$ (échantillonnage limite), (c) $\alpha=1.1$ (léger sous-échantillonnage).

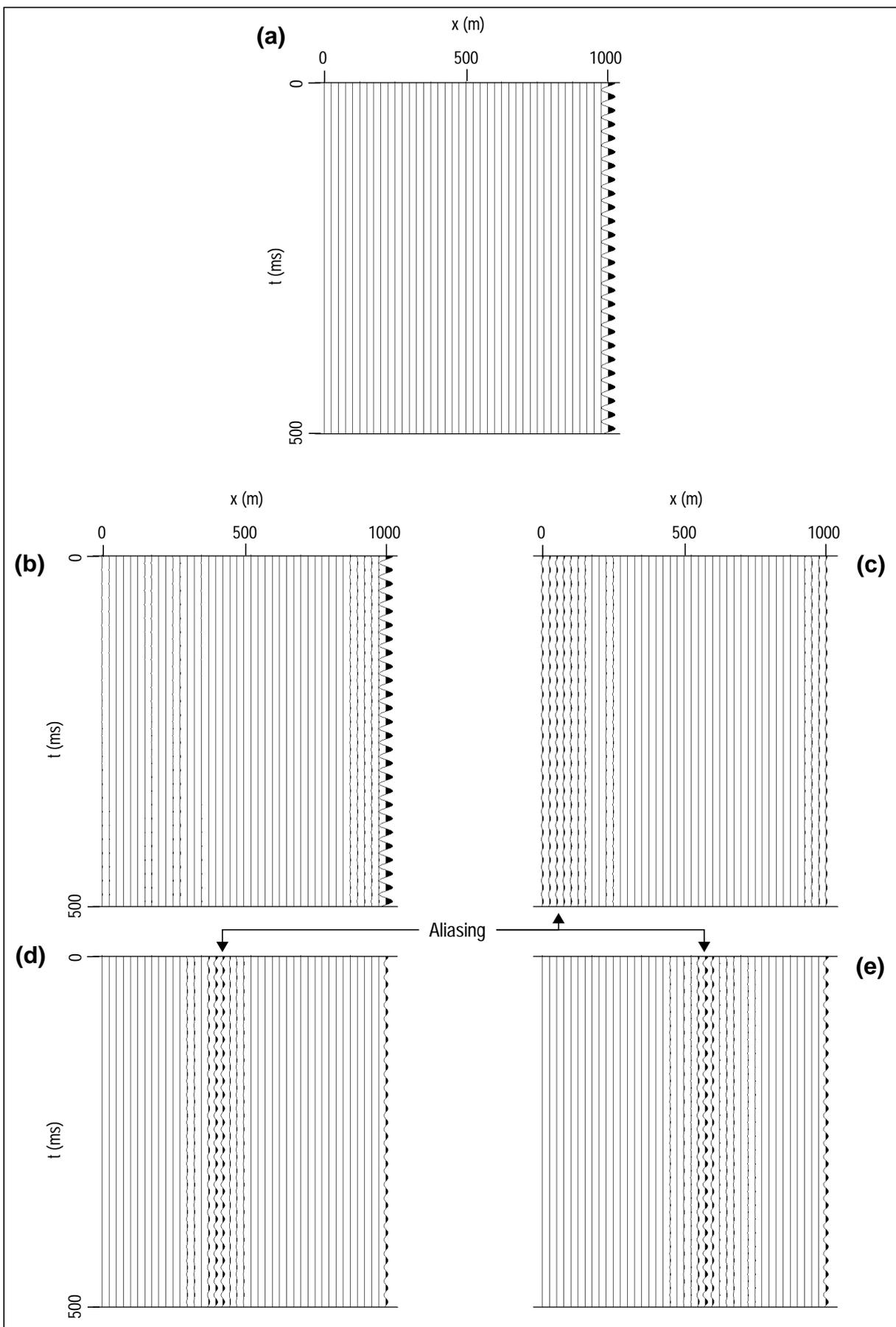


Figure 2.6 : (a) Collection de traces ($x_{min}=0$, $x_{max}=1000m$, $\Delta x=25m$). $f=50Hz$ sur trace 41. Reconstruction après PRD: (b) $\alpha=.8$, (c) $\alpha=1.0$, (d) $\alpha=1.2$, (e) $\alpha=1.5$. Pour $\alpha \geq 1.0$, la trace originelle est mal reconstruite et s'aliasse à des positions x qui dépendent de α .

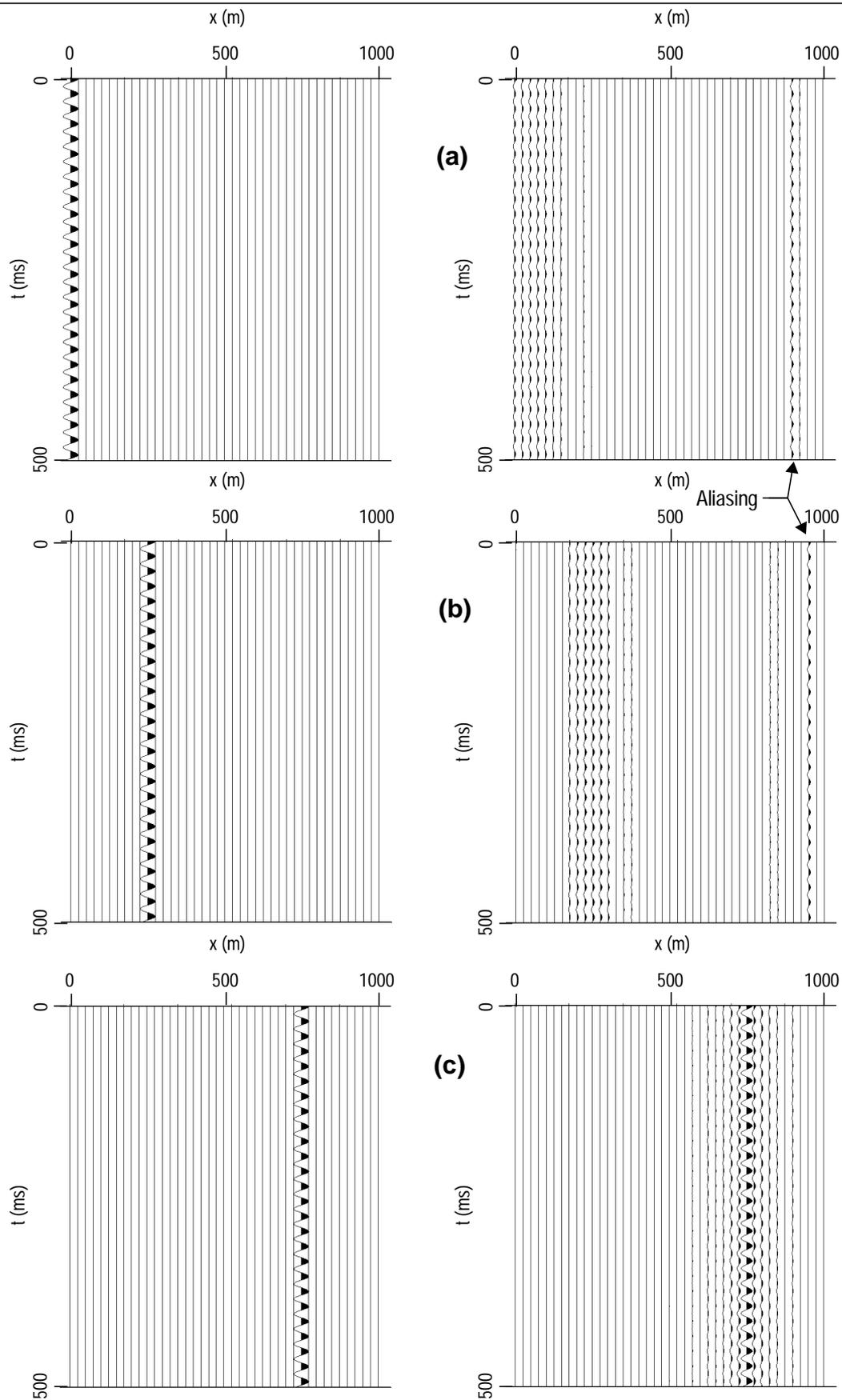


Figure 2.8 : Exemples de reconstructions après PRD, q sous-échantillonnée ($\alpha=1.2$), une seule trace non nulle dont la position varie: (a) trace 1, (b) trace 11, (c) trace 31, (voir fig. 2.6 pour la trace 41). La trace d'origine s'aliase sur un autre offset, sauf la trace 31, qui correspond à une zone du spectre de σ_{par} non repliée (fig. 2.7).

les hyperboles par la déflexion à un offset donné (E1.62). L'objectif est d'avoir un échantillonnage en T_X suffisamment dense pour que l'ensemble des trajectoires balaiant un temps inférieur ou égal à la longueur d'onde la plus courte du signal d'une trace du modèle à l'autre.

Pour un point quelconque (t,x), la trajectoire correspondante dans l'espace des modèles est (en combinant (E1.2), (E1.7) et (E1.62)):

$$G(t, x, T_X) = t - \frac{T_X}{2} + \frac{X^2 - \sqrt{T_X^4 V_e^4 + 4T_X^2 V_e^2 x^2 - 2T_X^2 V_e^2 X^2 + X^4}}{2T_X V_e^2} \quad (E2.2)$$

Cette équation, qui donne des trajectoires complexes, se simplifie fortement pour les cas $x=0$ et $x=X$:

$$\begin{aligned} G(t, x=0, T_X) &= t \\ G(t, x=X, T_X) &= t - T_X \end{aligned} \quad (E2.3)$$

Seules ces deux abscisses correspondent à des trajectoires rectilignes dans l'espace des modèles (fig. 2.9), contrairement aux cas de la LRD et de la PRD, où toutes les trajectoires sont rectilignes.

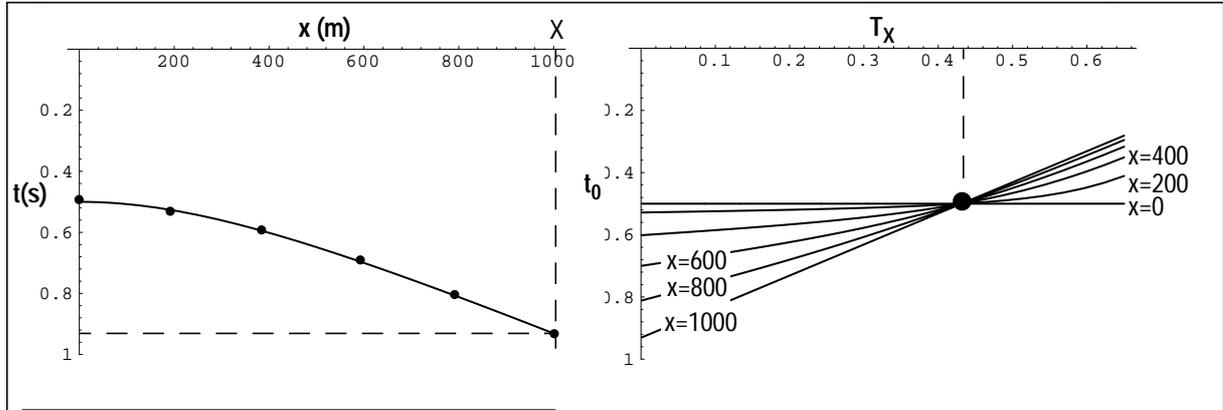


Figure 2.9 : Hyperboles P-SCAN: trajectoires dans le domaine (t_0, T_X) pour des points donnés dans le domaine (t, x) . Points choisis sur une hyperbole ($t_0=0.5s$, $T_X=0.438s$). Seules les abscisses $x=0$ et $x=X=1000m$ correspondent à des trajectoires rectilignes.

On vérifie que les gammes de pentes possibles dans le domaine (t_0, T_X) sont encadrées par les deux trajectoires rectilignes données par (E2.3): $\frac{\partial G}{\partial T_X} \text{max} - \frac{\partial G}{\partial T_X} \text{min} = 0 - (-1) = 1$

A l'intérieur de l'éventail défini par cette gamme, on veut qu'au maximum s'inscrive une longueur d'onde du signal entre 2 traces du modèle. Il faut donc que: $1 \cdot \Delta T_X \leq \frac{1}{f_{\text{max}}}$, f_{max} étant la fréquence maximum du signal. On retombe finalement sur la même règle pratique que pour les droites ou les paraboles, à savoir qu'au niveau de l'offset maximum (X ici), la «distance» entre deux hyperboles doit être inférieure à la longueur d'onde minimum que l'on veut représenter (fig. 2.1).

$$\boxed{\Delta T_X \leq \frac{1}{f_{\text{max}}}} \quad (E2.4)$$

De cette condition, on déduit les Δt_p trace par trace en appliquant l'équation (E1.62).

2.4 - Le pas d'échantillonnage en x - Aliasing spatial

2.4.1 - Théorie

L'espace le plus pratique pour étudier l'échantillonnage en x est l'espace des données (ω, x) dans lequel on étudie le spectre suivant x (nombre d'onde k_x), démarche abordée par plusieurs auteurs (Kostov, 1989).

En se donnant un modèle quelconque $\hat{u}(\omega, p)$, avec $p_{\min} \leq p \leq p_{\max}$, on prend la transformée de Fourier suivant x de l'opération de modélisation (E1.44):

$$\hat{d}(\omega, k_x) = \int_x \left(\int_p \hat{u}(\omega, p) \cdot e^{-i\omega p x} \cdot dp \right) \cdot e^{-ixk_x} \cdot dx$$

$$\hat{d}(\omega, k_x) = \int_p \hat{u}(\omega, p) \left(\int_x e^{-i(\omega p + k_x)x} \cdot dx \right) \cdot dp = \int_p \hat{u}(\omega, p) \cdot \delta(\omega p + k_x) \cdot dp$$

On en déduit que le spectre est nul en dehors de l'intervalle $-\omega p_{\max} \leq k_x \leq -\omega p_{\min}$. La théorie de l'échantillonnage nous conduit donc à prendre comme pas en x:

$$\Delta x \leq \frac{2\pi}{\omega(p_{\max} - p_{\min})} = \frac{1}{f(p_{\max} - p_{\min})} \quad (\text{E2.5})$$

Pour la PRD, on se ramène à une LRD par le changement de variable $X=x^2$. Cela conduit à la relation:

$$\Delta X = 2|x|\Delta x \leq \frac{1}{f(q_{\max} - q_{\min})} \quad (\text{E2.6})$$

Dans le cas de pas Δx constants, la condition porte sur l'offset maximum (la pente d'une parabole donnée est maximum à l'offset maximum):

$$\Delta x \leq \frac{1}{2f|x|_{\max}(q_{\max} - q_{\min})} \quad (\text{E2.7})$$

Lorsque la variable x est discrétisée, les opérateurs de convolution σ_{lin} et σ_{par} (équ. (E1.52)) s'expriment par une somme finie sur x à la place d'une intégrale:

$$\sigma_{\text{lin}}(\omega, p) = \sum_{x_j} e^{i\omega p x_j} \quad \sigma_{\text{par}}(\omega, p) = \sum_{x_j} e^{i\omega p x_j^2} \quad (\text{E2.8})$$

Ces opérateurs correspondent à la fois aux filtres appliqués à un modèle théorique (que l'on souhaite retrouver) pour donner une certaine image, et les signaux par lesquels on déconvolue cette image pour retrouver le modèle. On peut voir sur la figure 2.10 comment se modifient les opérateurs par la discrétisation de x.

On calcule les spectres des opérateurs, en supposant d'abord que p|q est d'extension infinie:

$$F\sigma_{\text{lin}}(\omega, k_p) = \int_p \left(\sum_{x_j} e^{2i\omega p x_j} \right) \cdot e^{-ipk_p} \cdot dp$$

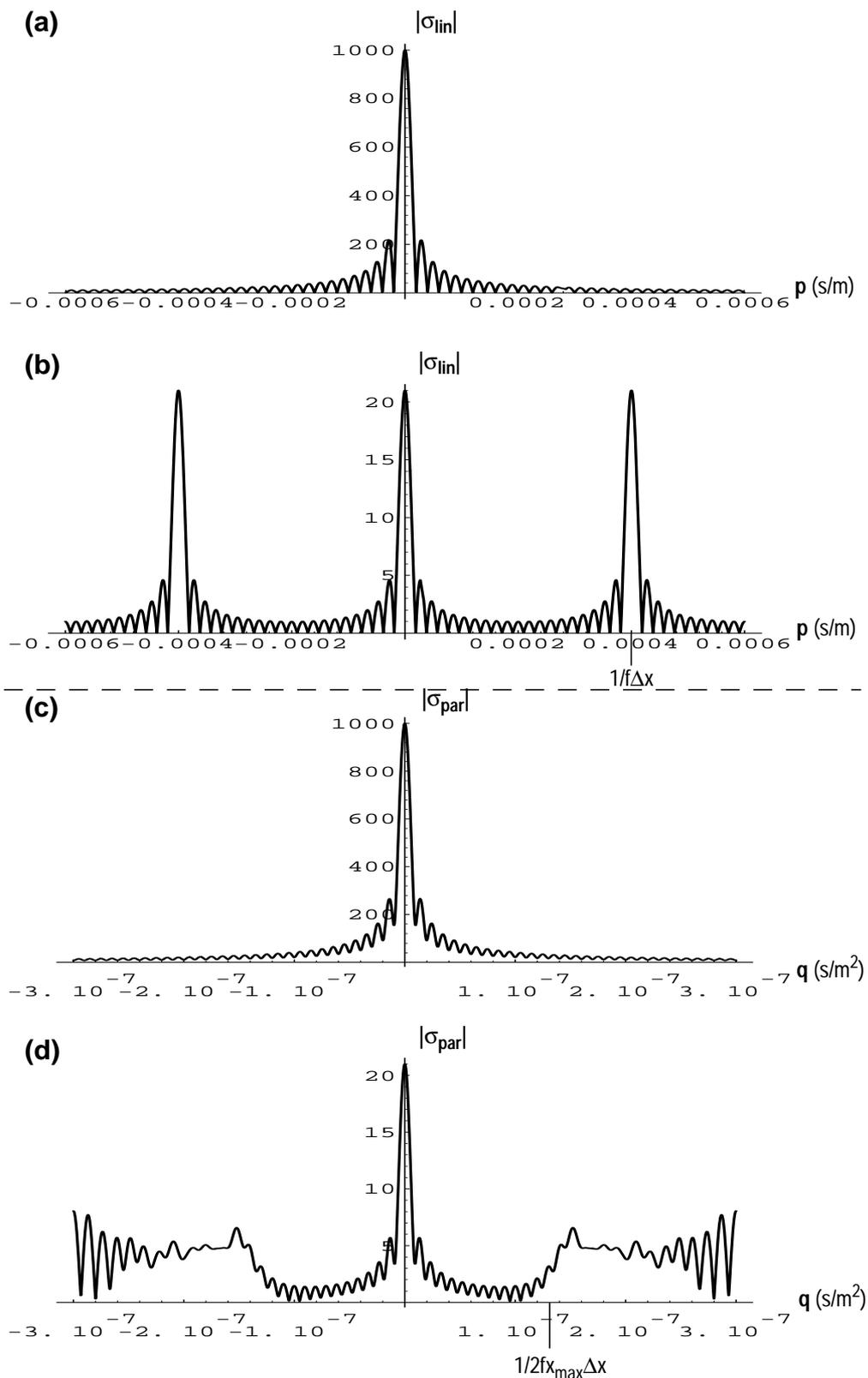


Figure 2.10 : Dégradation des opérateurs σ_{lin} et σ_{par} après discrétisation de x . (a) σ_{lin} pour x continu. (b) σ_{lin} pour x discrète. (c) σ_{par} pour x continue. (d) σ_{par} pour x discrète. ($f=50\text{Hz}$, $x_{min}=500\text{m}$, $x_{max}=1500\text{m}$, $\Delta x=50\text{m}$)

$$F\sigma_{lin}(\omega, k_p) = \sum_{x_j} \left(\int_p e^{i(\omega x_j - k_p)p} \cdot dp \right)$$

Le spectre de σ_{lin} est donc une succession de diracs:

$$F\sigma_{lin}(\omega, k_p) = \sum_{x_j} 2\pi\delta(\omega x_j - k_p) \quad (E2.9)$$

Pour la PRD, le même calcul donne:

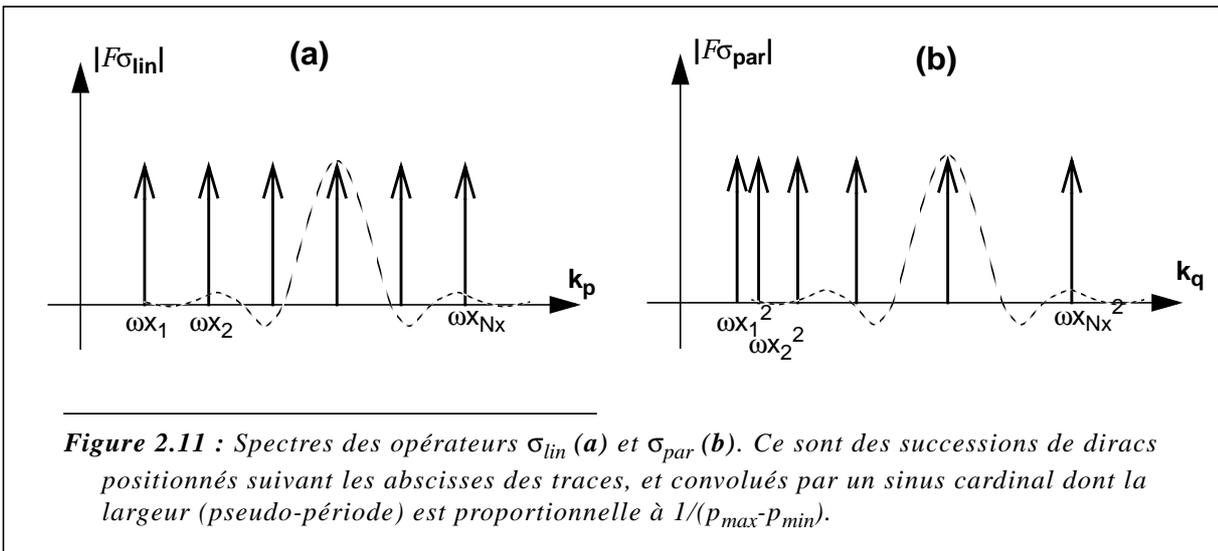
$$F\sigma_{par}(\omega, k_q) = \sum_{x_j} 2\pi\delta(\omega x_j^2 - k_q) \quad (E2.10)$$

Si $p|q$ n'est pas d'extension infinie, les spectres s'obtiennent en convoluant les résultats des équations (E2.9) et (E2.10) par le spectre de la fonction 'porte', qui est un sinus cardinal (ce sont les mêmes équations pour la variable q):

$$\Pi_{p_{max}, p_{min}}(p) = \begin{cases} 1 & \text{si } p_{min} \leq p \leq p_{max} \\ 0 & \text{sinon} \end{cases}$$

$$F\Pi_{p_{max}, p_{min}}(k) = \frac{e^{-ikp_{max}} - e^{-ikp_{min}}}{-ik} = \frac{2e^{-ik\left(\frac{p_{max} + p_{min}}{2}\right)} \sin\left(k\left(\frac{p_{max} - p_{min}}{2}\right)\right)}{k} \quad (E2.11)$$

Dans le cas limite d'une gamme de $p|q$ infinie, les spectres des opérateurs consistent en une succession de diracs, positionnés d'après les valeurs discrètes prises par x , et séparés par des plages nulles. Quand cette gamme diminue, ces diracs sont convolués par un sinus cardinal d'autant plus large que $(p_{max} - p_{min})$ est petit (fig. 2.11). Si l'échantillonnage en x est trop faible, les diracs sont trop espacés par rapport à la largeur du sinus cardinal, découvrant des plages sub-nulles, et rendant ainsi la déconvolution instable.



La dégradation du spectre de σ_{par} pour la PRD est illustré sur la figure 2.12: le problème apparaît d'abord dans la zone correspondant aux grands offsets.

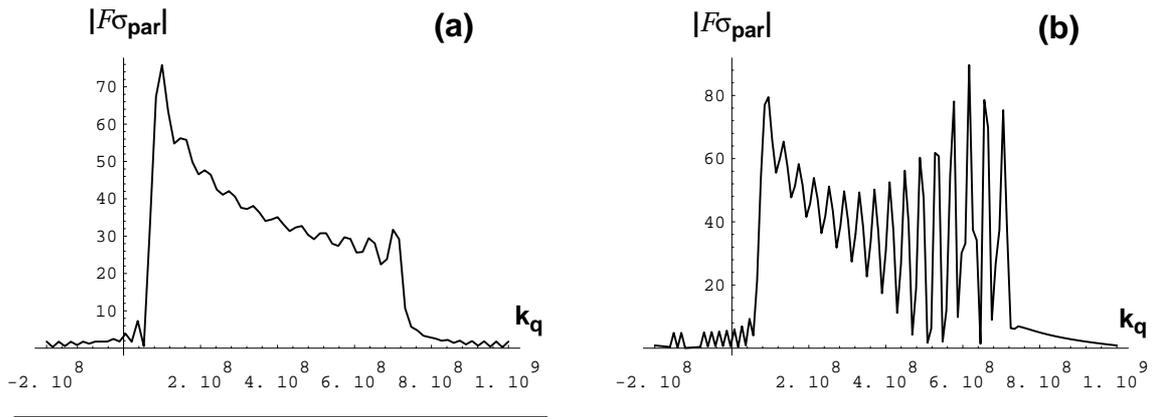


Figure 2.12 : dégradation du spectre de σ_{par} lorsque la variable x est sous-échantillonné. Mêmes paramètres que pour fig. 2.10d. (a) $q_{max}=-q_{min}=100\text{ms}/\text{km}^2$. (b) $q_{max}=-q_{min}=160\text{ms}/\text{km}^2$.

2.4.2 - Approche pratique

Le terme ‘aliasing’ désigne ici en fait deux problèmes liés mais différents dans leurs conséquences: le premier est celui de la stabilité numérique de la déconvolution (abordé au paragraphe précédent par l’intermédiaire du spectre des opérateurs), et le second est l’indétermination des solutions.

Turner (1990), pour la LRT, a abordé le problème de l’aliasing par un raisonnement graphique: c’est l’impossibilité de distinguer 2 événements monofréquentiels dont les pentes diffèrent de $1/f\Delta x$ (fig. 2.13). Si il existe effectivement un événement de pente p^* , lorsque l’on balaye les pentes l’aliasing arrive brutalement pour chaque pente $p=p^*+j/(f\Delta x)$, le long desquelles la sommation des amplitudes est maximum. Par contre, entre ces valeurs les interférences sont destructives.

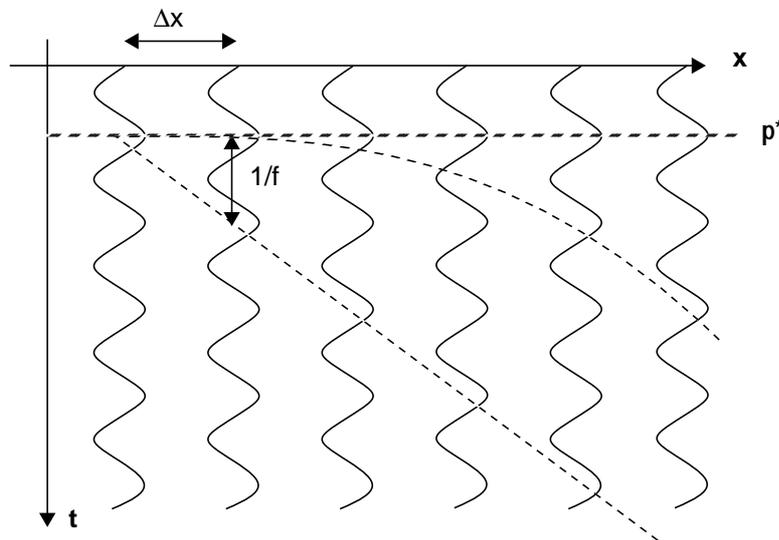


Figure 2.13 : pour une fréquence f donnée: en LRT, 2 événements dont les pentes diffèrent de $1/f\Delta x$ ne sont pas séparables car la sommation des amplitudes le long des 2 droites donne le même résultat. En PRT, aucune parabole (à part celle correspondant à la bonne courbure) ne somme constructivement toutes les traces.

2.4 -Le pas d'échantillonnage en x - Aliasing spatial

L'opérateur σ_{lin} devient ainsi parfaitement périodique (figure 2.10b), et la déconvolution instable. La solution de l'inversion, même stabilisée par un facteur d'amortissement (E1.35), contiendra tous les évènements possibles. La condition anti-aliasing s'écrit donc: $|p - p^*| \leq \frac{1}{f\Delta x}$

Cette dernière condition est moins restrictive que celle obtenue par (E2.5), car elle fait référence à la gamme des pentes explorées *relativement* à un évènement p^* , alors que précédemment on considérait la gamme totale de pentes. Cela signifie en pratique qu'un évènement particulier peut ne pas être aliasé alors que globalement la déconvolution est instable à cause de l'aliasing (cas par exemple d'un évènement $p^*=0$ (horizontal) et une gamme de pentes $p_{\text{max}}=-p_{\text{min}}=0.9/f\Delta x$: pour tout p , on a $|p-p^*|<1/f\Delta x$, mais par contre $|p_{\text{max}}-p_{\text{min}}|>1/f\Delta x$).

Pour la PRD, le problème est un peu différent. En effet, la pente d'une parabole varie tout au long des traces et seule la parabole parfaitement ajustée sur l'évènement q^* peut sommer toutes les amplitudes en phases (fig. 2.13). Néanmoins l'opérateur σ_{par} va tout même être dégradé (fig. 2.10d) dès lors que la pente locale des paraboles permettra une sommation constructive, c'est à dire si localement la pente est égale à $1/f\Delta x$.

La pente locale d'une parabole de courbure q valant $2qx$, c'est à l'offset maximum qu'elle est la plus forte. Si elle dépasse la valeur limite $1/f\Delta x$, alors il y a forcément une partie de la parabole le long de laquelle la sommation est constructive. La condition 'anti-aliasing' va donc s'écrire ici:

$$2|q - q^*||x|_{\text{max}} \leq \frac{1}{f\Delta x}$$

Finalement:

$$\boxed{\begin{aligned} |p - p^*| &\leq \frac{1}{f\Delta x} \\ |q - q^*| &\leq \frac{1}{2f|x|_{\text{max}}\Delta x} \end{aligned}} \quad (\text{E2.12})$$

A nouveau, le cas de la PRD apparaît à posteriori comme une extrapolation du cas de la LRD, mais le comportement des opérateurs est néanmoins assez différent (figure 2.10): alors que σ_{lin} est parfaitement périodique, σ_{par} présente un comportement plus ou moins erratique à partir d'un certain q , et l'indétermination est moins forte.

Pour terminer, on peut noter que le respect des conditions anti-aliasing dans les 2 domaines (données et modèles) entraîne un lien entre le nombre de traces des données (N_x) et le nombre de traces des modèles ($N_p|q$). Si l'on se place en conditions limites (pour la PRD, on suppose $x>0$ et $x_{\text{min}}=0$) données par les équations (E2.9) et (E2.10):

$$N_p = \frac{|p_{\text{max}} - p_{\text{min}}|}{\Delta p} = \frac{f(x_{\text{max}} - x_{\text{min}})}{f\Delta x} = N_x$$

$$N_q = \frac{|q_{\text{max}} - q_{\text{min}}|}{\Delta q} = \frac{f(x_{\text{max}}^2 - x_{\text{min}}^2)}{2f|x|_{\text{max}}\Delta x} = \frac{N_x}{2}$$

Dans ces conditions, pour la LRD le nombre d'inconnues (N_p) est égal au nombre de données (N_x), alors que pour la PRD il doit être deux fois plus petit. Lorsque l'on travaille avec un pas d'échantillonnage Δx constants dans la PRD, un échantillonnage suffisant aux grands offsets implique en effet un sur-échantillonnage aux faibles offsets.

2.4.3 - Exemple synthétique monofréquentiel

Afin de bien observer le comportement réel des transformées en conditions d'aliasing, le mieux est d'étudier un exemple simple. Pour cela nous reprenons les données de la figure 2.3, dont nous calculons les LRD et PRD pour différentes gammes de p et q . L'événement présent sur les données étant horizontal, les gammes de $p|q$ explorée sont symétriques ($p_{\min}=-p_{\max}$).

La géométrie des données implique les conditions anti-aliasing: $p_{\max}x_{\max} \leq 400\text{ms}$ pour la LRD, et $q_{\max}x_{\max}^2 \leq 200\text{ms}$ pour la PRD.

Sur la figure 2.14 se trouvent deux LRD et deux PRD de ces données, chacune en conditions non-aliasée et aliasée:

- pour la LRD, l'aliasing est très clair et consiste en une répétition de l'événement réel, qui n'est lui-même pas affecté (du moins pas sa forme, car son amplitude est 3 fois plus faible, le reste étant réparti sur les événements aliasés).
- pour la PRD l'aliasing se manifeste par l'apparition d'un signal diffus à partir du q limite d'aliasing. De plus l'événement central est affecté non seulement en amplitude, mais également en morphologie, contrairement au cas de la LRD: au lieu d'être représenté sur une seule trace, il s'étale maintenant sur les traces voisines. Aussi bien l'énergie diffusée que la 'défocalisation' peuvent être très gênante pour des traitements tels qu'un anti-multiple, qui nécessite une bonne séparation des événements.
- la répartition des énergies est cohérente avec la forme des opérateurs σ_{lin} et σ_{par} (fig. 2.10).

Sur la figure 2.15 sont tracées les normes L1 et L2 des LRD et PRD en fonction de p_{\max} et q_{\max} . On retrouve sur ces courbes les constatations faites auparavant, à savoir:

- pour la LRD, la transition est brutale lorsqu'on arrive à la limite d'aliasing: la norme L2 est quasi-constante de part et d'autre de la limite et chute brutalement au niveau de cette limite. La norme L1 varie peu, avec toutefois une rupture au niveau de la limite.
- pour la PRD, la norme L2 est à peu près constante, puis commence à décroître régulièrement à partir de l'aliasing. La norme L1 quant à elle croît régulièrement.

Ce comportement des transformées résulte bien de la résolution du problème inverse en moindres carrés. Parmi toutes les solutions possibles, c'est celle d'énergie minimum qui est choisie, et non pas forcément celle que l'on souhaiterait. Si il y a trois événements équivalents pouvant expliquer les données (cas de la LRD aliasée fig. 2.14), alors l'inversion répartit l'énergie sur ces trois événements.

2.4.4 - Exemple synthétique 2

Un signal monofréquentiel est très pratique pour bien comprendre les causes et les conséquences de l'aliasing, mais il faut maintenant observer ce qui se passe avec un signal sismique classique composé d'une gamme de fréquence normale.

La limite d'aliasing dépendant de la fréquence, il est clair qu'il n'y aura pas de transition nette comme précédemment, mais certainement une dégradation progressive des transformations.

Soit une collection de traces avec un unique événement horizontal habillé par un ricker de fréquence centrale $f_c=30\text{Hz}$ (fig. 2.16).

Nous calculons les PRD de ces données avec différents q_{\max} (fig 2.17). Comme repère, le q_{\max} d'aliasing pour la fréquence centrale du signal ($f_c=30\text{Hz}$) vaut, d'après l'équation (E2.12), 333ms/km^2 . On observe très bien les figures d'aliasing en 'ailes de papillons', et également la déformation de l'événement central (moins bonne focalisation) quand la transformée est aliasée: une partie de son

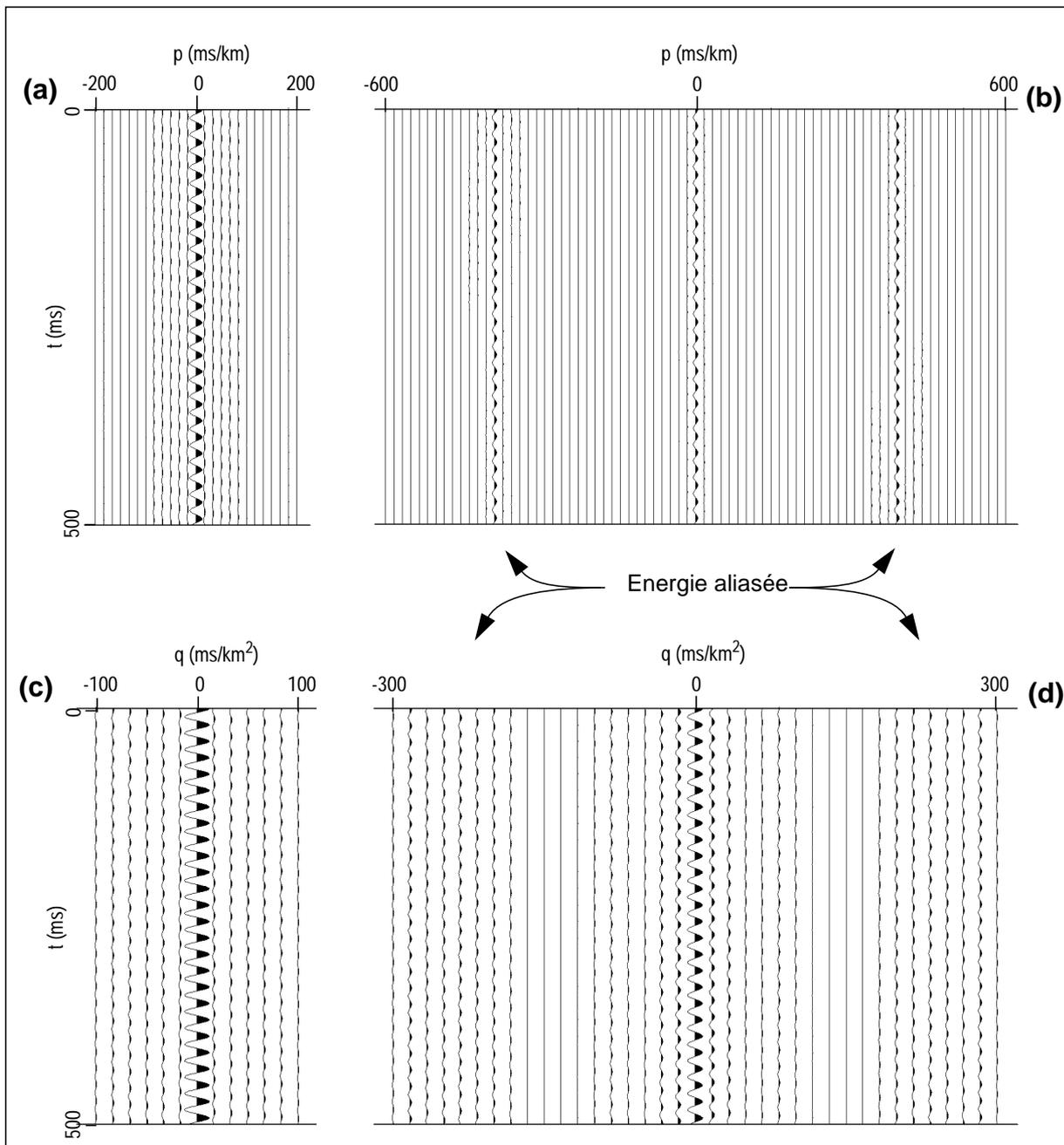


Figure 2.14 : LRD et PRD des données fig. 2.3 pour différents p_{max} . (a) LRD $p_{max}=200\text{ms/km}$. (b) LRD $p_{max}=600\text{ms/km}$ ($\text{gain}(b)=3 \times \text{gain}(a)$). (c) PRD $q_{max}=100\text{ms/km}^2$. (d) PRD $q_{max}=300\text{ms/km}^2$.

énergie se reporte sur les traces latérales. Ces PRD contiennent toute l'information nécessaire pour bien reconstruire les données, mais celle-ci se répartit différemment.

Les variations des normes L1 et L2 des deux transformées (LRD et PRD) en fonctions des gammes explorées (fig 2.18) sont identiques à celles de l'exemple du paragraphe précédent: la norme L2 diminue et la norme L1 augmente lorsque les gammes augmentent.

Des PRD, nous ne conservons maintenant que les traces centrales (valeurs de $|q|$ inférieures à 150ms/km^2) pour reconstruire les données (fig 2.17 à droite)². Dans la première décomposition ($q_{max}=150\text{ms/km}^2$) toute l'information est contenue autour de la trace $q=0$, et on peut reconstruire les données de départ. Par contre, dans les deux autres ($q_{max}=300$ et 500ms/km^2) la reconstruction est imparfaite: on constate une perte d'amplitude le long de l'évènement, en fonction de l'offset (surtout aux grands offsets). L'énergie manquante est en fait située sur les traces aliasées des PRD, traces que l'on a

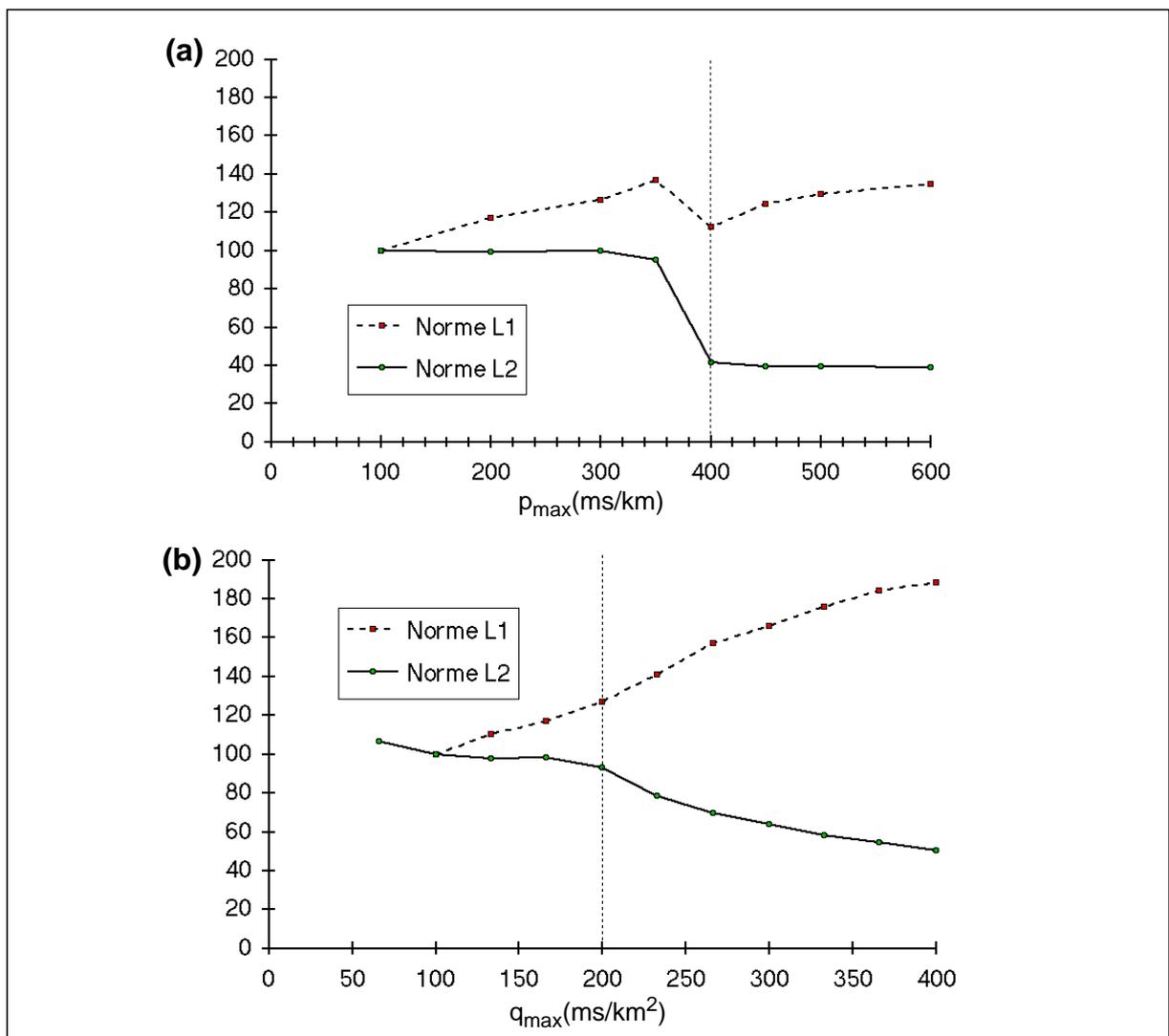


Figure 2.15 : Evolution des normes L1 et L2 des décompositions des données de la figure 2.3 lorsque la gamme en p/q explorée varie. (a) LRD. (b) PRD

éliminées. Ces problèmes d'amplitude sont particulièrement sensibles si l'on souhaite faire une étude AVO sur les données, la perte d'amplitude variant en effet avec l'offset.

2.4.5 - Comment remédier à l'aliasing spatial

Pour la PRD, l'aliasing spatial est influencé par quatre paramètres (équ. (E2.12)):

- la fréquence
- l'offset maximum
- le pas d'échantillonnage spatial
- la gamme de courbures explorées

Les solutions les plus évidentes seraient donc:

² Ce processus, aujourd'hui couramment utilisé, permet de séparer primaires et multiples sur un CMP après correction NMO: si la différence de courbure (donc de vitesse) entre primaires et multiples est suffisante, ceux-ci vont apparaître sur la PRD dans des zones bien distinctes. On peut alors les séparer par un mute et reconstruire uniquement les primaires (ou uniquement les multiples, que l'on soustrait du CMP initial).

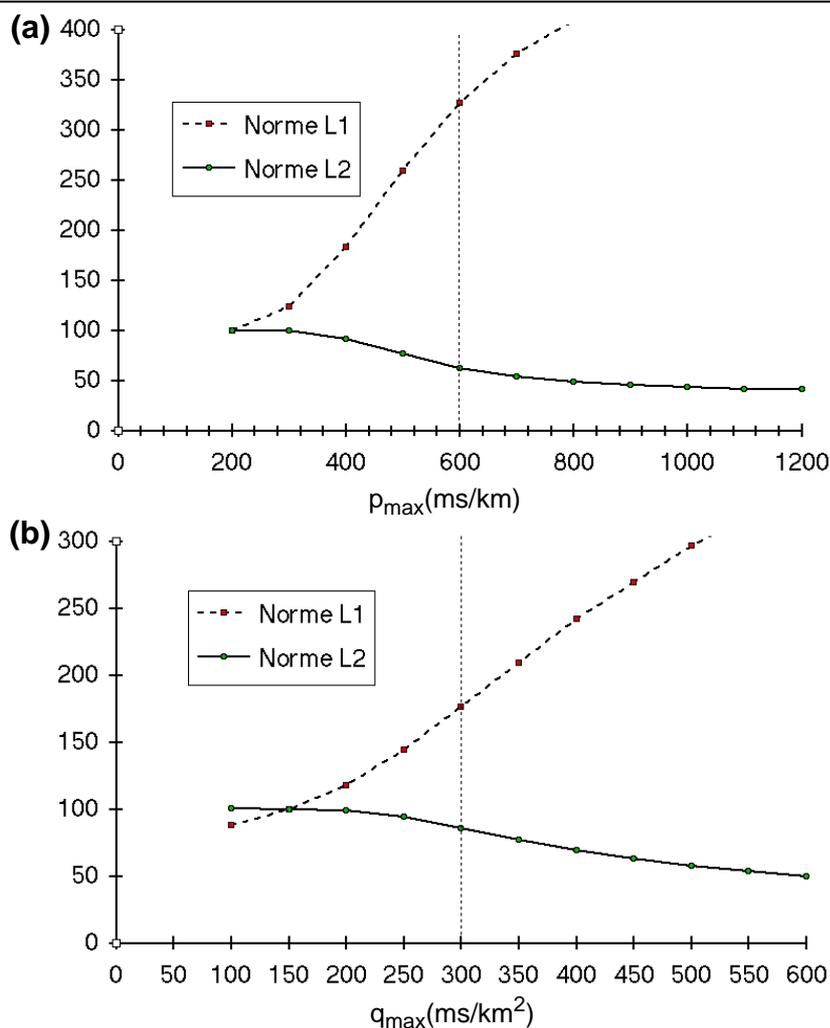
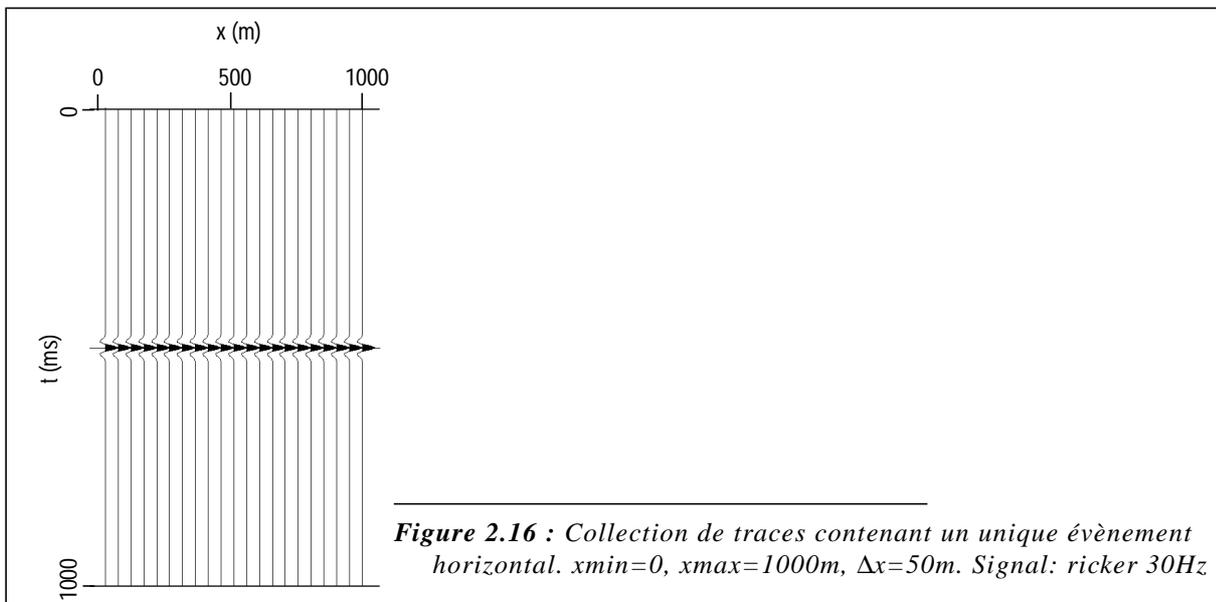


Figure 2.18 : Evolution des normes L1 et L2 des décompositions des données de la figure 2.16 lorsque la gamme en p/q explorée varie. (a) LRD. (b) PRD

- l'élimination des hautes fréquences sur les grands offsets, qui sont les plus aliasées. Mais cette solution a bien sûr l'inconvénient majeur d'éliminer de l'information, ce qui n'est jamais satisfaisant.
- la création de nouvelles traces par interpolation afin de réduire le pas d'échantillonnage spatial, en utilisant bien sûr des interpolateurs qui réduisent l'aliasing (Spitz, 1991).

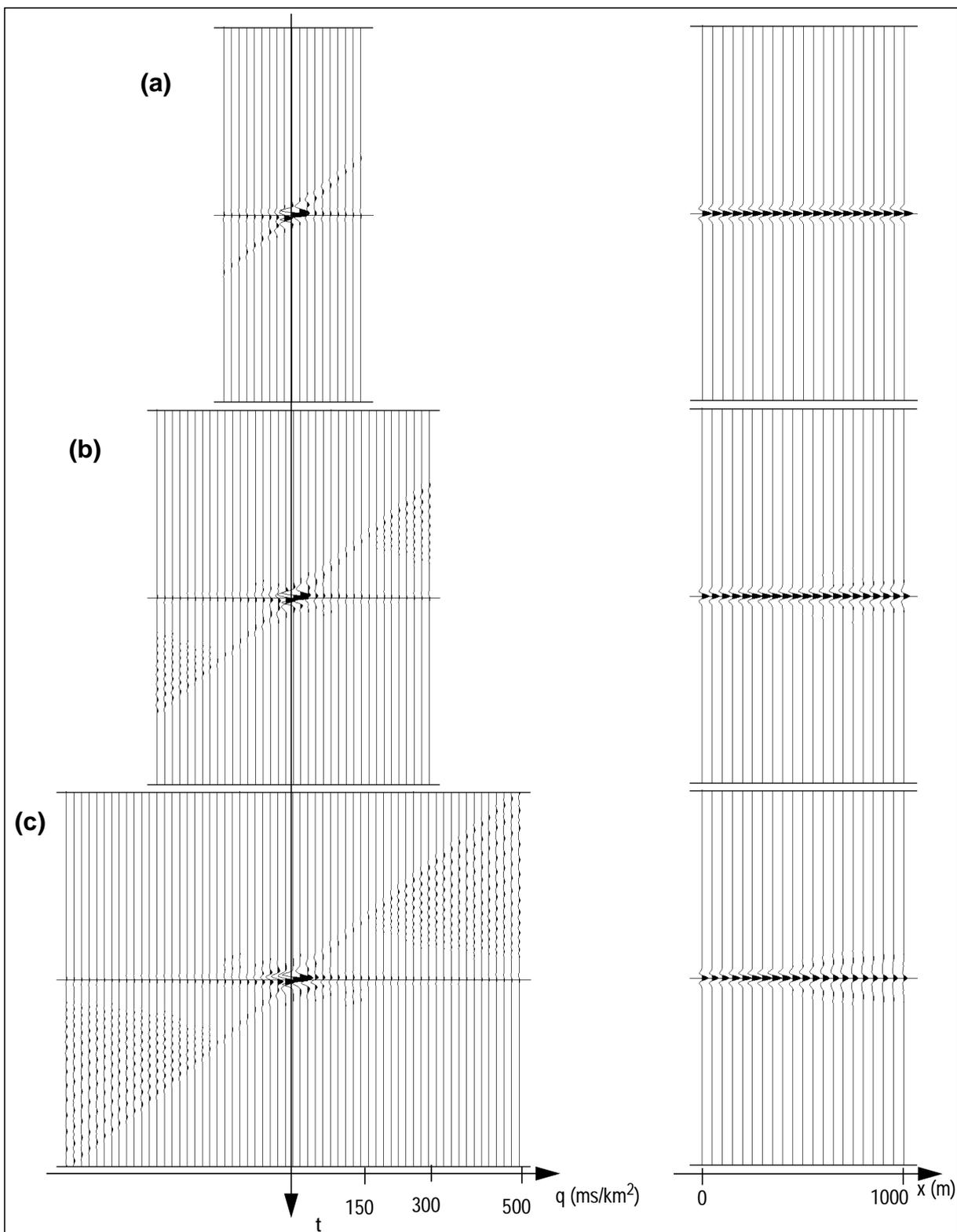


Figure 2.17 : A gauche: PRD des données fig. 2.16 avec différentes valeurs de q_{max} : (a) $150\text{ms}/\text{km}^2$. (b) $300\text{ms}/\text{km}^2$. (c) $500\text{ms}/\text{km}^2$. On observe la déformation de l'évènement central en même temps qu'apparaissent les figures d'aliasing.

A droite: les reconstructions correspondantes après avoir muté les traces $|q| > 150\text{ms}/\text{km}^2$. Il y a une perte d'amplitude en fonction de l'offset en cas d'aliasing.

- une gamme de courbures la plus faible possible, tout en respectant ce qui est contenu dans les données: en effet si on choisit une gamme trop faible excluant certains évènements, on peut effectivement réduire l'aliasing, mais l'inversion ne peut pas expliquer ces évènements, ce qui engendre des artefacts pouvant être aussi gênants que ceux engendrés par l'aliasing.

D'autres solutions moins évidentes sont envisageables:

Inversion de l'opérateur (LL*)

L'aliasing se traduisant numériquement par une instabilité de l'opérateur $(\mathbf{L}^*\mathbf{L})^{-1}$, il est quelquefois conseillé de calculer l'opérateur $(\mathbf{L}\mathbf{L}^*)^{-1}$ qui doit être plus stable dans ce cas (Kostov, 1990), et d'appliquer l'inverse à droite au lieu de l'inverse à gauche (voir (E1.34)).

D'un point de vue théorique, il est équivalent d'utiliser l'un ou l'autre opérateur, qui doivent conduire à la même solution. L'inverse à droite ne résout ici que le problème de l'instabilité, pas celui de l'indétermination.

D'un point de vue pratique, l'inconvénient principal en PRD est que seule la matrice $\mathbf{L}^*\mathbf{L}$ est de Toeplitz, $\mathbf{L}\mathbf{L}^*$ étant quelconque (Darche, 1990; en LRD les opérateurs sont tous deux de Toeplitz si Δx est constant). Inverser $\mathbf{L}^*\mathbf{L}$ est donc beaucoup plus rapide (par l'algorithme de Levinson) qu'inverser $\mathbf{L}\mathbf{L}^*$.

La différence peut se faire au niveau du facteur d'amortissement utilisé, qui peut être d'autant plus faible que l'opérateur est stable. Néanmoins, la présence de bruit dans les données réelles conduit à *toujours* introduire un facteur d'amortissement (Tarantola, 1987), ce qui réduit la portée de cet effet en pratique.

Finalement on obtient des solutions très semblables quel que soit l'opérateur utilisé (Kostov, 1990).

Décompositions parcimonieuses

Une autre voie plus intéressante est suggérée par les courbes de la figure 2.18: sous l'effet de l'aliasing spatial, la norme L1 de la solution augmente, alors que la norme L2 diminue. Peut-être faut-il alors régulariser l'inversion par la norme L1.

Pour une donnée \mathbf{d} , au lieu de choisir la solution de norme L2 minimum:

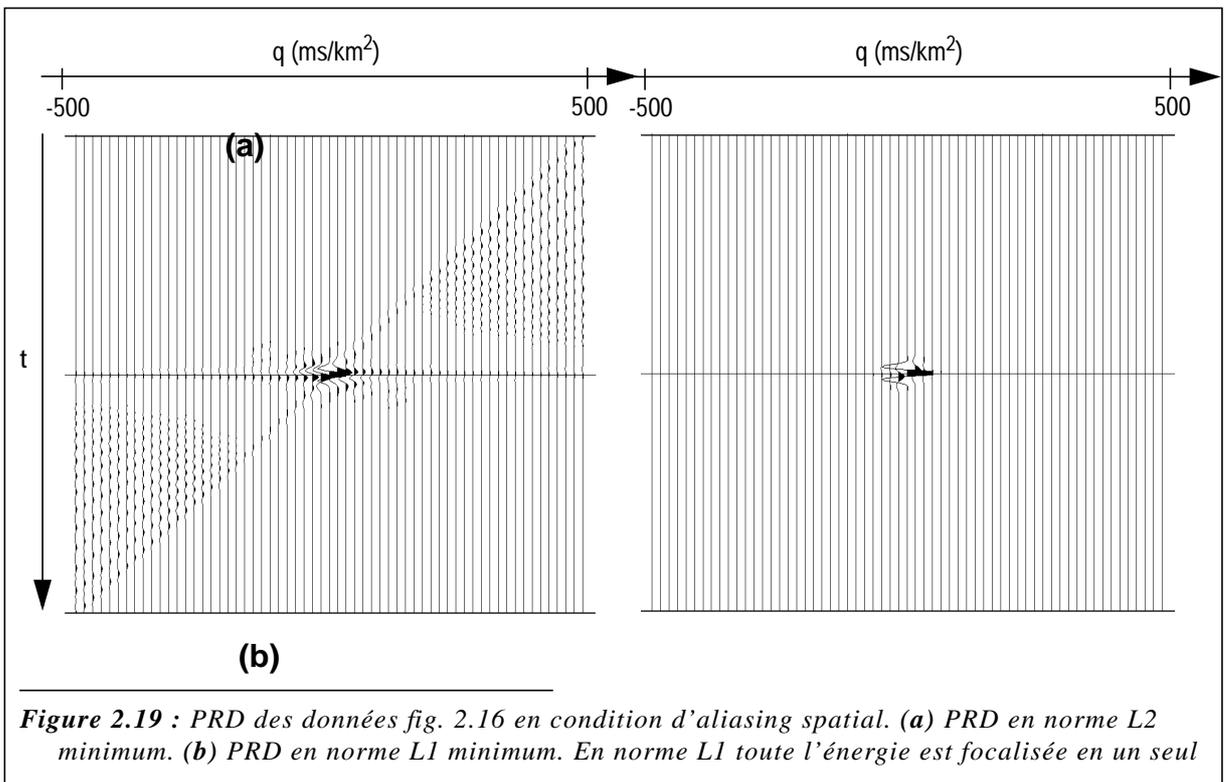
$$\min_{\mathbf{u}} (\|\mathbf{d} - \mathbf{L}\mathbf{u}\|_2^2 + \varepsilon \|\mathbf{u}\|_2^2) \quad \|\mathbf{u}\|_2^2 = \iint_{x_0 t_0} \mathbf{u}^2(t_0, x_0) dt_0 dx_0$$

on choisit la solution de norme L1 minimum (Nichols, 1994):

$$\min_{\mathbf{u}} (\|\mathbf{d} - \mathbf{L}\mathbf{u}\|_2^2 + \varepsilon \|\mathbf{u}\|_1) \quad \|\mathbf{u}\|_1 = \iint_{x_0 t_0} |\mathbf{u}(t_0, x_0)| dt_0 dx_0$$

La figure 2.19 contient deux PRD des données de la figure 2.16, calculées avec une gamme de q telle que l'aliasing est parfaitement visible sur la PRD en norme L2. Sur l'autre PRD, calculée en norme L1, les figures d'aliasing ont disparues et l'évènement central est beaucoup mieux focalisé. On sait d'une manière générale qu'une des propriétés de la norme L1 est de rassembler au maximum l'énergie d'un évènement en un seul point de la transformée: cela concerne entre autre la partie aliasée de l'évènement.

On obtient en fait des résultats comparables, à des degrés divers, en utilisant à la place de la norme L2 toute autre minimisation qui tend à concentrer l'énergie en des points privilégiés de la transformée. D'une manière générale, on appelle ces solutions des solutions *parcimonieuses* (voir chapitre 4).



2.5 - Conclusions

Pour les cas usuels de transformées de Radon, l'échantillonnage des variables temps t et t_0 n'est pas problématique: les fréquences maximums rencontrées sur les données et sur le modèle sont liés par le stretch factor (égal à 1.0 pour les familles de droites, de paraboles, et d'hyperboles P-SCAN).

Les pas d'échantillonnage pour p (LRD) et q (PRD) se déduisent de l'ouverture ($x_{\max} - x_{\min}$) et de la fréquence maximum des données. Le sur-échantillonnage de ces variables n'est pas utile pour la qualité de la décomposition (ni même souhaitable du point de vue de la stabilité de l'inversion). Le sous-échantillonnage ne permet pas de représenter correctement les données les plus HF et les plus proches des bords du domaine spatial.

Le pas d'échantillonnage spatial Δx devrait être déterminé à l'acquisition des données suivant les gammes de pentes p ou de courbures q susceptibles d'être contenues dans les données. Une fois ce pas fixé, si les gammes de pentes ou de courbures dépassent une valeur critique l'inversion est instable à cause de l'aliasing spatial. Pour autant, tous les événements ne sont pas forcément aliasés: la condition d'aliasing pour un événement donné porte non pas sur la gamme de pentes (ou de courbures) totale, mais sur la différence maximum entre les pentes utilisées et la pente de l'évènement.

L'aliasing spatial n'empêche pas une décomposition des données (les résidus sont faibles) mais se traduit par une dispersion de l'énergie dans le modèle loin de la position attendue.

3 - Le Problème inverse

3.1 - Introduction

Nous nous intéressons dans ce chapitre au(x) formalisme(s) du problème inverse appliqué à la décomposition de Radon, i.e. la résolution du problème $\mathbf{d} = \mathbf{L}\mathbf{u}$.

La formulation des moindres carrés généralisés (par exemple Tarantola, 1987) introduit le modèle à priori, ainsi que les matrices de covariances sur les données et sur le modèle, qui décrivent les incertitudes sur les données observées et sur le modèle à priori.

Le formalisme de Bayes repose quant à lui sur les notions de probabilités et d'état d'information sur les différentes variables du système. Les moindres carrés généralisés constituent en fait un cas particulier d'inversion bayésienne.

L'application pratique de ces concepts n'est pas forcément aisée -parfois ni même possible- mais ils permettent d'avoir une bonne compréhension des phénomènes en jeu. Ils montrent notamment comment introduire de l'information à priori dans l'inversion, et donnent une meilleure compréhension de la signification de certains paramètres (tel que le facteur d'amortissement).

La décomposition de Radon est posée sous une forme proche des moindres carrés généralisés, ce qui permet entre autres de la résoudre par la minimisation de formes quadratiques. Compte tenu de la taille des systèmes en jeu, on préfère une méthode itérative telle que le gradient conjugué (dans le domaine temps) aux méthodes directes. Le domaine temps est par ailleurs plus adapté que le domaine de Fourier dès lors que la décomposition n'est pas effectuée en moindres carrés simples (norme L2), à cause de l'application de pondération variables en temps notamment.

La décomposition en norme L2 est étudiée plus en détail, notamment la signification et le choix du facteur d'amortissement. L'influence des variations de couverture (dus aux mutes par exemple) est prise en considération.

Enfin nous étudions l'utilisation des filtres rho (rho-filter, Claerbout, 1985) pour accélérer la convergence du gradient conjugué. Alors que les filtres rho usuels sont déduits des transformées continues et infinies, nous proposons pour la LRD (et la PRD) un filtre rho déduit des transformées discrètes et finies.

3.2 - Moindres carrés généralisés

Le problème inverse, que l'on appelle **Décomposition de Radon**, consiste, pour une donnée $\mathbf{d}(t,x)$ à trouver un modèle $\mathbf{u}(t_0,x_0)$ qui satisfasse l'équation de modélisation:

$$\mathbf{d} \in \text{ID}, \text{ trouver } \mathbf{u} \in \text{IU} \quad / \quad \mathbf{d} = \mathbf{L}\mathbf{u} \quad (\mathbf{E3.1})$$

Ce problème est linéaire, et en travaillant avec un nombre fini de variables il s'agit d'un système linéaire matriciel à résoudre.

Thorson (1984), considérant la LRD et la HRD, a mis en évidence que ce problème était mal posé, en ce sens qu'il admet en général une infinité de solutions (sous-détermination):

- L'ouverture limitée en x laisse la possibilité d'expliquer un événement, même parfait, par plusieurs événements proches.
- Pour les transformées paraboliques et hyperboliques des CMP, toutes les courbes se rejoignent et s'aplatissent vers $x=0$, ce qui rend difficile la séparation des événements sur les offsets faibles.
- On est souvent amené à prendre un nombre d'inconnues (proportionnel au nombre de valeurs de x_0) supérieur au nombre d'équations (proportionnel au nombre de valeurs de x).

- Les données peuvent être spatialement aliasées, ce qui oblige en pratique à avoir plus d'inconnues que d'équations.

certaines données peuvent par ailleurs ne pas admettre de solutions (sur-détermination), s'il s'agit par exemple d'un bruit aléatoire, difficilement modélisable par un nombre limité d'évènements organisés. Ainsi, comme cela est souvent le cas avec des mesures physiques, le problème est à la fois sous- et sur-déterminé.

Il convient donc de définir des critères supplémentaires pour choisir une solution. Thorson (1984) en a exploré plusieurs: solution d'énergie minimale, d'entropie minimale, etc...

Tarantola (1987) a présenté une généralisation de la résolution classique du problème en norme L2, ou moindres carrés (équ. (E1.36)), valable pour tout problème inverse (y compris non linéaire). On cherche un modèle \mathbf{u} qui minimise la forme quadratique:

$$\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d}_{\text{obs}} - \underline{\mathbf{L}}\mathbf{u}, \underline{\mathbf{C}}_{\mathbf{d}}^{-1}(\mathbf{d}_{\text{obs}} - \underline{\mathbf{L}}\mathbf{u})) + \frac{1}{2}(\mathbf{u} - \mathbf{u}_{\text{p}}, \underline{\mathbf{C}}_{\mathbf{u}}^{-1}(\mathbf{u} - \mathbf{u}_{\text{p}})) \quad (\text{E3.2})$$

\mathbf{d}_{obs} est la donnée observée, \mathbf{u}_{p} est un modèle à priori. $\underline{\mathbf{C}}_{\mathbf{d}}$ est la matrice de covariance sur les données, qui décrit une incertitude gaussienne sur \mathbf{d}_{obs} . $\underline{\mathbf{C}}_{\mathbf{u}}$ est la matrice de covariance sur le modèle, qui décrit l'incertitude gaussienne sur le modèle \mathbf{u}_{p} fourni à priori. La minimisation de φ conduit au modèle \mathbf{u} 'le plus probable' (minimiser φ par rapport à un modèle revient en effet à maximiser la probabilité d'avoir observé \mathbf{d}_{obs} pour ce modèle). On qualifie cette résolution de **moindres carrés généralisés**. Les moindres carrés sont étroitement liés à la notion de statistique gaussienne, où toutes les erreurs et incertitudes doivent être décrites de cette façon.

Les matrices de covariance (et donc leurs inverses) sont symétriques. $\underline{\mathbf{C}}_{\mathbf{d}}$ doit décrire à la fois le **bruit de mesure** (l'erreur sur l'observation de \mathbf{d}) et le **bruit de modélisation**, c'est à dire les écarts entre la réalité et ce que l'on est capable de modéliser avec l'opérateur $\underline{\mathbf{L}}$ (en décomposition hyperbolique par exemple, on cherche à expliquer les données par une superposition d'hyperboles d'amplitudes constantes alors que les indicatrices ne sont ni parfaitement hyperboliques, ni d'amplitude constante).

Lemme: La solution des moindres carrés généralisés est:

$$\mathbf{u}_2 = (\underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \underline{\mathbf{L}} + \underline{\mathbf{C}}_{\mathbf{u}}^{-1})^{-1} (\underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \mathbf{d}_{\text{obs}} + \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}_{\text{p}}) \quad (\text{E3.3})$$

Démonstration:

Le minimum de φ est atteint lorsque son gradient s'annule. Calculons celui-ci:

$$\begin{aligned} \varphi(\mathbf{u}) &= \frac{1}{2}((\mathbf{d}_{\text{obs}}, \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \mathbf{d}_{\text{obs}}) - 2(\mathbf{d}_{\text{obs}}, \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \underline{\mathbf{L}}\mathbf{u}) + (\underline{\mathbf{L}}\mathbf{u}, \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \underline{\mathbf{L}}\mathbf{u}) + (\mathbf{u}, \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}) \\ &\quad - 2(\mathbf{u}_{\text{p}}, \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}) + (\mathbf{u}_{\text{p}}, \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}_{\text{p}})) \\ \varphi(\mathbf{u}) &= \frac{1}{2}((\mathbf{d}_{\text{obs}}, \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \mathbf{d}_{\text{obs}}) - 2(\underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \mathbf{d}_{\text{obs}}, \mathbf{u}) + (\underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \underline{\mathbf{L}}\mathbf{u}, \mathbf{u}) + (\underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}, \mathbf{u}) \\ &\quad - 2(\underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}_{\text{p}}, \mathbf{u}) + (\mathbf{u}_{\text{p}}, \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}_{\text{p}})) \end{aligned}$$

$$\nabla \varphi(\mathbf{u}) = -\underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \mathbf{d}_{\text{obs}} + \underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} \underline{\mathbf{L}}\mathbf{u} + \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u} - \underline{\mathbf{C}}_{\mathbf{u}}^{-1} \mathbf{u}_{\text{p}}$$

D'où:

$$\nabla \varphi(\mathbf{u}) = -\underline{\mathbf{L}}^* \underline{\mathbf{C}}_{\mathbf{d}}^{-1} (\mathbf{d}_{\text{obs}} - \underline{\mathbf{L}}\mathbf{u}) + \underline{\mathbf{C}}_{\mathbf{u}}^{-1} (\mathbf{u} - \mathbf{u}_{\text{p}}) \quad (\text{E3.4})$$

L'identification de cette dernière quantité à **0** conduit au résultat désiré.

Dans la pratique, surtout pour des problèmes contenant un très grand nombre d'inconnues, la détermination des matrices de covariance est un problème difficile, pratiquement aussi difficile que la résolution du problème lui-même.

3.3 - Formalisme bayésien

L'approche bayésienne de l'inversion est fondée sur les probabilités, ou plus exactement sur les notions d'état d'information et de degré de vraisemblance. Il n'est pas question de présenter en détail la théorie, avec ses différentes interprétations (on en trouvera un exposé complet et clair dans la thèse de Duijndam (1987)), mais de rappeler les principaux résultats utilisés ici.

Les données et les modèles sont considérées comme étant des réalisations de variables aléatoires. Notons $p(\cdot)$ la fonction de densité de probabilité (**pdf**) d'une variable (ou d'un vecteur de variables) aléatoire. La pdf d'une variable décrit **l'état d'information** sur cette variable. Les pdf les plus couramment utilisées sont les distributions gaussiennes, exponentielle, ou uniforme. Une distribution de dirac traduit une information totale sur une variable (on connaît parfaitement la valeur prise par la variable, qui est la position du dirac).

La **pdf composée** de deux vecteurs \mathbf{x} et \mathbf{y} est notée $p(\mathbf{x}, \mathbf{y})$. La **pdf conditionnelle** de \mathbf{x} pour \mathbf{y} donné $p(\mathbf{x}|\mathbf{y})$ est la pdf décrivant l'état d'information sur \mathbf{x} pour une réalisation particulière (fixée) de \mathbf{y} .

Si les deux variables sont indépendantes, il est clair que: $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y})$ et $p(\mathbf{x}|\mathbf{y}) = p(\mathbf{x})$.

Dans le cas général où les variables ne sont pas indépendantes, la pdf composée de \mathbf{x} et \mathbf{y} est le produit de la pdf de \mathbf{y} par la pdf conditionnelle de \mathbf{x} pour \mathbf{y} donné (et réciproquement):

$$\begin{aligned} p(\mathbf{x}, \mathbf{y}) &= p(\mathbf{x}|\mathbf{y})p(\mathbf{y}) \\ &= p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) \end{aligned} \tag{E3.5}$$

Un exemple simple couramment donné pour expliquer cette relation est celui du cycliste: la probabilité qu'il puisse faire l'aller-retour (x, y) entre deux villes A et B est égale à la probabilité qu'il puisse aller (x) de A à B, *et* qu'il puisse revenir (y) de B à A compte tenu du fait qu'il vient d'effectuer le parcours A->B et est donc très fatigué.

On déduit de la relation précédente la fameuse **loi de Bayes**:

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})} \tag{E3.6}$$

Cette dernière relation¹ est à la base de l'inversion bayésienne. En effet, pour un problème inverse que l'on peut exprimer sous la forme générale:

$$\mathbf{d} = \mathbf{f}(\mathbf{u}) + \mathbf{n} \tag{E3.7}$$

¹. Pour reprendre l'exemple du cycliste, et dans un souci de réalisme, une variable z représentant le dopage du cycliste pourra être introduite. La loi de Bayes donne alors: $p(z|(x, y)) = p((x, y)|z)p(z)/p(x, y)$, ce qui se traduit par: la probabilité qu'un cycliste ayant effectué l'aller-retour A<->B soit dopé est égale à la probabilité de faire A<->B en étant dopé divisée par la réussite moyenne observée sur le parcours A<->B, le tout pondéré par la proportion moyenne de dopés chez les cyclistes.

où \mathbf{d} sont les données observées, \mathbf{u} est le modèle cherché qui peut expliquer \mathbf{d} par l'intermédiaire de l'opérateur de modélisation f (quelconque). \mathbf{n} contient les erreurs de modélisation et le bruit de mesure. La loi de Bayes permet d'exprimer la pdf conditionnelle de \mathbf{u} pour \mathbf{d} donné:

$$p(\mathbf{u}|\mathbf{d}) = \frac{p(\mathbf{d}|\mathbf{u})p(\mathbf{u})}{p(\mathbf{d})} \quad (\text{E3.8})$$

\mathbf{d} étant fixé, le dénominateur est un terme constant qui ne sert qu'à la normalisation. La pdf conditionnelle de \mathbf{d} pour \mathbf{u} donné, en tant que fonction de \mathbf{u} , est appelée la **fonction de vraisemblance (likelihood function)**:

$$l(\mathbf{u}) = p(\mathbf{d}|\mathbf{u}) \quad (\text{E3.9})$$

\mathbf{d} étant donné, à une certaine réalisation \mathbf{u} correspond une réalisation du bruit \mathbf{n} égale à $(\mathbf{d} - f(\mathbf{u}))$. Si \mathbf{n} est indépendant de $f(\mathbf{u})^2$ et a pour pdf p_n , alors:

$$l(\mathbf{u}) = p(\mathbf{d}|\mathbf{u}) = p_n(\mathbf{d} - f(\mathbf{u})) \quad (\text{E3.10})$$

Parmi toutes les réalisations possibles de \mathbf{u} , celle qui maximise la fonction de vraisemblance est l'**estimateur de vraisemblance maximum (Maximum Likelihood Estimator=MLE)**. Cet estimateur ne conduit pas forcément à une solution unique.

$$\text{MLE: } \max_{\mathbf{u}} [l(\mathbf{u}) = p(\mathbf{d}|\mathbf{u}) = p_n(\mathbf{d} - f(\mathbf{u}))] \quad (\text{E3.11})$$

Dans la relation (E3.8), la pdf de \mathbf{u} traduit l'état d'information sur \mathbf{u} indépendamment de \mathbf{d} . C'est la **pdf à priori**. Si on ne dispose pas d'information à priori sur \mathbf{u} , on doit prendre la pdf qui traduit l'état d'information nul, par exemple une distribution uniforme³. Dans l'approche classique de l'inversion, ce terme est appelé la régularisation.

Finalement la pdf conditionnelle de \mathbf{u} pour \mathbf{d} donné est appelée la **pdf à posteriori**:

$$p(\mathbf{u}|\mathbf{d}) = \frac{p_n(\mathbf{d} - f(\mathbf{u}))p(\mathbf{u})}{p(\mathbf{d})} \quad (\text{E3.12})$$

La maximisation de cette pdf à posteriori fournit un autre estimateur de \mathbf{u} (qui n'est pas forcément unique non plus): l'**estimateur de vraisemblance maximum à posteriori (Maximum A Posteriori Estimator =MAP)**:

$$\text{MAP: } \max_{\mathbf{u}} [p(\mathbf{u}|\mathbf{d})] = \max_{\mathbf{u}} [p_n(\mathbf{d} - f(\mathbf{u}))p(\mathbf{u})] \quad (\text{E3.13})$$

Quant à l'**estimateur de Bayes** proprement dit, c'est la moyenne de tous les \mathbf{u} possibles, pondérés par leur pdf à posteriori:

$$\text{Estimateur de Bayes: } \mathbf{u}_{\text{Bayes}} = \int \mathbf{u} p(\mathbf{u}|\mathbf{d}) d\mathbf{u} \quad (\text{E3.14})$$

² L'indépendance entre \mathbf{n} et $f(\mathbf{u})$ est discutable dès lors que \mathbf{n} contient le bruit de modélisation, ce qui est le cas dans le contexte des Décompositions de Radon. Toutefois on fait comme si cette condition était respectée.

³ l'état d'information nul ne se traduit pas forcément par une distribution uniforme

Ce dernier estimateur est sans doute le plus fiable (du moins le plus complet) des trois décrits ici. Malheureusement il est inutilisable en pratique dans la plupart des cas, car nécessitant de parcourir tout l'espace des modèles en calculant les pdf à posteriori. L'estimateur MAP est donc le plus souvent utilisé.

La résolution de notre problème inverse en minimisant la fonction coût (E3.2) peut être considérée comme de une estimation MAP, avec pour pdf les distributions gaussiennes suivantes:

$$\begin{aligned}
 p_{\mathbf{n}}(\mathbf{n}) &= \frac{1}{\sqrt{(2\pi)^{n_d} \|\underline{\mathbf{C}}_D\|}} e^{-\frac{1}{2}(\mathbf{n}, \underline{\mathbf{C}}_D^{-1} \mathbf{n})} \\
 p(\mathbf{u}) &= \frac{1}{\sqrt{(2\pi)^{n_u} \|\underline{\mathbf{C}}_U\|}} e^{-\frac{1}{2}(\mathbf{u} - \mathbf{u}_p, \underline{\mathbf{C}}_U^{-1} (\mathbf{u} - \mathbf{u}_p))}
 \end{aligned}
 \tag{E3.15}$$

n_d et n_u sont les dimensions respectives de l'espace des données et de l'espace des modèles, $\underline{\mathbf{C}}_D$ et $\underline{\mathbf{C}}_U$ sont des matrices de covariance. Le bruit \mathbf{n} est décrit comme gaussien, de moyenne nulle, dont les variances et covariances sont contenues dans la matrice $\underline{\mathbf{C}}_D$. La pdf de \mathbf{u} définit un à priori sur le modèle, supposé égal à \mathbf{u}_p ($p(\mathbf{u})$ est maximum pour $\mathbf{u}=\mathbf{u}_p$), mais cette connaissance à priori est entachée d'une certaine incertitude décrite par les variances et covariances de la matrice $\underline{\mathbf{C}}_U$ (une variance élevée signifie que la plage des valeurs probables pour un échantillon du modèle est plus large).

Calculer l'estimation MAP de \mathbf{u} revient à maximiser la pdf à posteriori $p(\mathbf{u}|\mathbf{d})$. Le dénominateur dans (E3.12) étant constant, il suffit de maximiser la quantité $p_{\mathbf{n}}(\mathbf{d} - \underline{\mathbf{L}}\mathbf{u})p(\mathbf{u})$. Compte tenu de la forme exponentielle de ces pdf, on en prend le logarithme, qui devra être maximum lui aussi. Les termes constants s'additionnent après l'application du logarithme et n'influent pas sur la position du maximum. Il faut donc maximiser:

$$\max_{\mathbf{u}} \left(-\frac{1}{2}(\mathbf{d} - \underline{\mathbf{L}}\mathbf{u}, \underline{\mathbf{C}}_D^{-1} (\mathbf{d} - \underline{\mathbf{L}}\mathbf{u})) - \frac{1}{2}(\mathbf{u} - \mathbf{u}_p, \underline{\mathbf{C}}_U^{-1} (\mathbf{u} - \mathbf{u}_p)) \right)
 \tag{E3.16}$$

On retombe donc sur la même formulation que les moindres carrés généralisés de (E3.2).

Lorsque le modèle à priori est nul ($\mathbf{u}_p=\mathbf{0}$) (cas le plus courant pour les décompositions de Radon), le terme de régularisation portant sur le modèle traduit implicitement un à priori sur \mathbf{u} , supposé nul avec un certain degré d'incertitude, avec pour conséquence d'introduire un biais vers $\mathbf{0}$ dans la solution finale.

Quant à la résolution en norme L2 (poids diagonaux uniformes, (E1.36)), elle suppose implicitement non seulement que le bruit sur les données et les valeurs sur le modèle obéissent à des distributions gaussiennes, mais aussi que les espérances pour chaque échantillon sont partout les mêmes. Cela peut sembler raisonnable pour le bruit (encore qu'après correction de divergence sphérique le niveau de bruit augmente avec le temps, par exemple), mais moins pour le modèle.

3.4 - Cadre général de résolution

On cherche à résoudre le problème $\mathbf{d} = \mathbf{L}\mathbf{u}$ en minimisant la fonction coût:

$$\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d} - \mathbf{L}\mathbf{u}, \mathbf{P}_D(\mathbf{d} - \mathbf{L}\mathbf{u}))_{ID} + \frac{1}{2}(\mathbf{u}, \mathbf{P}_U\mathbf{u})_{IU} \quad (\text{E3.17})$$

\mathbf{P}_D et \mathbf{P}_U étant des poids appliqués respectivement aux résidus et au modèle. Ils sont symétriques, définis et positifs, de telle sorte que les opérateurs $(\cdot, \mathbf{P}\cdot)$ sont également des produits scalaires. Pour faire le lien entre cette fonction coût et la formulation des moindres carrés généralisés (E3.2), on pose simplement: $\mathbf{C}_D^{-1} = \mathbf{P}_D$ et $\mathbf{C}_U^{-1} = \mathbf{P}_U$. Par ailleurs, si le modèle à priori \mathbf{u}_p n'est pas nul, on pose: $\mathbf{u}' = \mathbf{u} - \mathbf{u}_p$ et $\mathbf{d}'_{\text{obs}} = \mathbf{d}_{\text{obs}} - \mathbf{L}\mathbf{u}_p$, et, à condition que \mathbf{L} soit linéaire ($\mathbf{L}\mathbf{u} = \mathbf{L}(\mathbf{u}' + \mathbf{u}_p) = \mathbf{L}\mathbf{u}' + \mathbf{L}\mathbf{u}_p$), l'équation (E3.2) devient:

$$\Phi(\mathbf{u}') = \varphi(\mathbf{u}' + \mathbf{u}_p) = \frac{1}{2}(\mathbf{d}'_{\text{obs}} - \mathbf{L}\mathbf{u}', \mathbf{C}_D^{-1}(\mathbf{d}'_{\text{obs}} - \mathbf{L}\mathbf{u}')) + \frac{1}{2}(\mathbf{u}', \mathbf{C}_U^{-1}\mathbf{u}') ,$$

permettant de se ramener à l'expression (E3.17).

On ne traitera ici que les poids diagonaux, les poids non diagonaux rendant les problèmes beaucoup trop lourds à implémenter pour les Décompositions de Radon, bien qu'il n'y ait aucun obstacle théorique. \mathbf{P}_D et \mathbf{P}_U étant les inverses des matrices de covariance \mathbf{C}_D et \mathbf{C}_U , des éléments non diagonaux nuls signifient que les échantillons sont décorrélés les uns des autres (**ce qui n'est pas à priori une évidence et doit être justifié au cas par cas**).

La solution à laquelle on arrive par la minimisation de (E3.17) est:

$$\mathbf{u} = (\mathbf{L}^* \mathbf{P}_D \mathbf{L} + \mathbf{P}_U)^{-1} \mathbf{L}^* \mathbf{P}_D \mathbf{d} \quad (\text{E3.18})$$

On note que des poids proportionnels à l'identité conduisent à la solution classique en norme L2.

On rencontre souvent une autre façon d'écrire ces équations, en répartissant les poids sur les deux termes des produits scalaires:

$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{W}_D \mathbf{d} - \mathbf{W}_D \mathbf{L} \mathbf{u}\|_{L_2}^2 + \frac{1}{2} \|\mathbf{W}_U \mathbf{u}\|_{L_2}^2 \quad (\text{E3.19})$$

$$\mathbf{P}_D = (\mathbf{W}_D)^2 \quad \mathbf{P}_U = (\mathbf{W}_U)^2$$

Minimiser cette dernière quantité revient à opérer des changements de variables sur l'espace des données et sur l'espace des modèles:

$$\mathbf{d}' = \mathbf{W}_D \mathbf{d} \quad \mathbf{u}' = \mathbf{W}_U \mathbf{u} \quad \mathbf{L}' = \mathbf{W}_D \mathbf{L} \mathbf{W}_U^{-1} \quad (\text{E3.20})$$

et à résoudre le nouveau système en moindres carrés classiques:

$$\mathbf{d}' = \mathbf{L}' \mathbf{u}' \quad (\text{E3.21})$$

La solution \mathbf{u}' n'a donc pas d'énergie dans le noyau de l'opérateur $\mathbf{L}' = \mathbf{W}_D \mathbf{L} \mathbf{W}_U^{-1}$, et la partie des données \mathbf{d}' orthogonale à l'espace image de cet opérateur reste dans les résidus. Exprimé dans les

variables d'origine, \mathbf{u} n'a donc pas d'énergie dans le noyau de $\tilde{\mathbf{L}}'\tilde{\mathbf{W}}_U^{-1} = \tilde{\mathbf{W}}_D\tilde{\mathbf{L}}\tilde{\mathbf{W}}_U^{-2}$, et les résidus de \mathbf{d} se situent dans l'espace image de $\tilde{\mathbf{W}}_D\tilde{\mathbf{L}}' = \tilde{\mathbf{W}}_D^2\tilde{\mathbf{L}}\tilde{\mathbf{W}}_U^{-1}$.

La norme L2 se caractérise par des poids diagonaux uniformes. L'application d'autres poids permet de:

- Modifier le noyau de l'opérateur de modélisation \mathbf{L} (par le poids \mathbf{W}_U), changeant ainsi la forme de la solution (puisque celle-ci n'a pas de composante dans le noyau).
- Favoriser (resp. défavoriser) certaines zones du modèle, en appliquant des poids plus faibles (resp. plus forts) aux endroits où on souhaite voir une concentration d'énergie (resp. une énergie faible).
- Sur les données, de traiter le cas des spikes éventuels, en appliquant aux échantillons où ils sont présents un poids plus faible.

3.5 - Résolution par gradient conjugué

3.5.1 - Problèmes quadratiques (poids constants)

La fonction coût à minimiser est:

$$\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}, \tilde{\mathbf{P}}_d(\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u})) + \frac{1}{2}(\mathbf{u}, \tilde{\mathbf{P}}_u\mathbf{u}) \quad (\text{E3.22})$$

On réécrit cette fonction sous la forme utilisée dans l'annexe sur le gradient conjugué:

$$\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d}, \tilde{\mathbf{P}}_d\mathbf{d}) - (\mathbf{b}, \mathbf{u}) + \frac{1}{2}(\mathbf{u}, \tilde{\mathbf{A}}\mathbf{u}) \quad (\text{E3.23})$$

en posant:

$$\mathbf{b} = \tilde{\mathbf{L}}^*\tilde{\mathbf{P}}_d\mathbf{d} \quad \tilde{\mathbf{A}} = \tilde{\mathbf{L}}^*\tilde{\mathbf{P}}_d\tilde{\mathbf{L}} + \tilde{\mathbf{P}}_u \quad (\text{E3.24})$$

On applique ensuite les formules du gradient du gradient conjugué. Le 'résidu' (au sens des formules du GC) en particulier, qui est l'opposé du gradient, s'écrit:

$$\begin{aligned} \mathbf{r}_b &= \mathbf{b} - \tilde{\mathbf{A}}\mathbf{u} = \tilde{\mathbf{L}}^*\tilde{\mathbf{P}}_d(\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}) - \tilde{\mathbf{P}}_u\mathbf{u} \\ &= \tilde{\mathbf{L}}^*\tilde{\mathbf{P}}_d\mathbf{r} - \tilde{\mathbf{P}}_u\mathbf{u} \end{aligned} \quad (\text{E3.25})$$

où \mathbf{r} est ici le résidu au sens habituel, par rapport aux données \mathbf{d} .

L'implémentation choisie est donc la suivante:

- (iv) Initialisations: $\mathbf{u} \leftarrow \mathbf{u}_0$ (en général $\mathbf{u}_0 = \mathbf{0}$)
 $\mathbf{r} \leftarrow \mathbf{d} - \mathbf{L}\mathbf{u}_0$
- (v) Calcul du gradient: $\mathbf{g} \leftarrow \mathbf{L}^*\mathbf{P}_d\mathbf{r} - \mathbf{P}_u\mathbf{u}$
- (vi) Facteur de conjugaison: $\gamma \leftarrow (\mathbf{g}, \mathbf{g})$
 1ère itération: $\mathbf{g}_c \leftarrow \mathbf{g}$
 Itérations suivantes: $\mathbf{g}_c \leftarrow \mathbf{g} + \frac{\gamma}{\gamma_{-1}}\mathbf{g}_c$
 stocker le produit scalaire: $\gamma_{-1} \leftarrow \gamma$
- (vii) Synthétique de la direction de descente: $\mathbf{l}\mathbf{g}_c \leftarrow \mathbf{L}\mathbf{g}_c$
- (viii) Recherche linéaire du minimum (voir ci-dessous), et mise à jour $\mathbf{u} \leftarrow \mathbf{u} + \alpha\mathbf{g}_c$
 $\mathbf{r} \leftarrow \mathbf{r} - \alpha\mathbf{l}\mathbf{g}_c$
- (ix) aller à ii) ou arrêter.

D'après l'équation (EA.5), le coefficient de recherche linéaire le long de \mathbf{g}_c s'obtient par:

$$\alpha = \frac{(\mathbf{r}_b, \mathbf{g}_c)}{(\mathbf{g}_c, \mathbf{A}\mathbf{g}_c)} = \frac{(\mathbf{b} - \mathbf{A}\mathbf{u}, \mathbf{g}_c)}{(\mathbf{g}_c, \mathbf{A}\mathbf{g}_c)}$$

D'une manière plus générale, les recherches multi-linéaires impliquent le calcul de termes similaires à ces derniers numérateurs et dénominateurs (équ. (EA.19)). Ceux-ci s'écrivent:

$$\begin{aligned} (\mathbf{b} - \mathbf{A}\mathbf{u}, \mathbf{g}_c) &= (\mathbf{L}^*\mathbf{P}_d\mathbf{d}, \mathbf{g}_c) - ((\mathbf{L}^*\mathbf{P}_d\mathbf{L} + \mathbf{P}_u)\mathbf{u}, \mathbf{g}_c) & (\mathbf{g}_c, \mathbf{A}\mathbf{g}_c) &= (\mathbf{g}_c, (\mathbf{L}^*\mathbf{P}_d\mathbf{L} + \mathbf{P}_u)\mathbf{g}_c) \\ &= (\mathbf{P}_d\mathbf{d}, \mathbf{L}\mathbf{g}_c) - (\mathbf{P}_d\mathbf{L}\mathbf{u}, \mathbf{L}\mathbf{g}_c) - (\mathbf{P}_u\mathbf{u}, \mathbf{g}_c) & &= (\mathbf{L}\mathbf{g}_c, \mathbf{P}_d\mathbf{L}\mathbf{g}_c) + (\mathbf{g}_c, \mathbf{P}_u\mathbf{g}_c) \\ &= (\mathbf{P}_d\mathbf{r}, \mathbf{L}\mathbf{g}_c) - (\mathbf{P}_u\mathbf{u}, \mathbf{g}_c) & &= (\mathbf{P}_d\mathbf{L}\mathbf{g}_c, \mathbf{L}\mathbf{g}_c) + (\mathbf{P}_u\mathbf{g}_c, \mathbf{g}_c) \\ &= (\mathbf{r}, \mathbf{P}_d\mathbf{l}\mathbf{g}_c) - (\mathbf{u}, \mathbf{P}_u\mathbf{g}_c) & &= (\mathbf{l}\mathbf{g}_c, \mathbf{P}_d\mathbf{l}\mathbf{g}_c) + (\mathbf{g}_c, \mathbf{P}_u\mathbf{g}_c) \end{aligned}$$

Donc finalement:

$$\alpha = \frac{(\mathbf{r}, \mathbf{P}_d\mathbf{l}\mathbf{g}_c) - (\mathbf{u}, \mathbf{P}_u\mathbf{g}_c)}{(\mathbf{l}\mathbf{g}_c, \mathbf{P}_d\mathbf{l}\mathbf{g}_c) + (\mathbf{g}_c, \mathbf{P}_u\mathbf{g}_c)} \quad (\text{E3.26})$$

Temps de calcul

Chaque itération réclame l'application une fois chacun des opérateurs \mathbf{L}^* (calcul du gradient, étape (ii)) et \mathbf{L} (modélisation de la direction de descente, étape (iv)). Par ailleurs, il faut calculer un produit scalaire dans la formule de conjugaison (étape (iii)) et 4 autres dans la recherche linéaire (étape (iv)). Il y a de plus quatre applications de poids (si on en utilise) dans les produits scalaires.

Concernant l'espace mémoire, on a besoin de 2 tableaux de la dimension de \mathbf{u} (\mathbf{g} et \mathbf{g}_c), et de 1 tableau de la dimension de \mathbf{d} ($\mathbf{l}\mathbf{g}_c$). \mathbf{g} et $\mathbf{l}\mathbf{g}_c$ ne sont jamais utilisés au même moment, ce qui diminue l'espace mémoire nécessaire.

Si N_t , N_d , N_u sont respectivement le nombre d'échantillon temps, le nombre de traces dans les données, et le nombre de traces dans le modèle: \mathbf{L} et \mathbf{L}^* réclament de l'ordre de $N_t.N_d.N_u$ opérations; un produit scalaire de l'ordre de $N_t.N_d$ (espaces des données) ou $N_t.N_u$ (espace des modèles); si les poids utilisés sont diagonaux, leur application nécessite également $N_t.N_d$ ou $N_t.N_u$ opérations (dans le cas

contraire cela peut aller jusqu'au carré de ces valeurs). Ce sont donc les opérateurs \underline{L} et \underline{L}^* qui sont les plus coûteux en temps de calcul (80% du temps total en moyenne).

Il n'y a aucune inversion de matrice à effectuer, d'où l'intérêt de la méthode si l'on peut atteindre la convergence en peu d'itérations.

3.5.2 - Préconditionnement

Si on souhaite appliquer un opérateur preconditionnement \underline{Q} au problème, les formules (EA.14) indiquent que seuls les calculs du gradient et du facteur de conjugaison sont modifiés, menant à l'implémentation suivante:

- (i) Initialisations: $\mathbf{u} \leftarrow \mathbf{u}_0$ (en général $\mathbf{u}_0 = \mathbf{0}$)
 $\mathbf{r} \leftarrow \mathbf{d} - \underline{L}\mathbf{u}_0$
- (ii) Calcul du gradient vrai: $\mathbf{g} \leftarrow \underline{L}^*\mathbf{P}_d\mathbf{r} - \mathbf{P}_u\mathbf{u}$
- (iii) Préconditionnement: $\mathbf{g}_p \leftarrow \underline{Q}\mathbf{g}$
- (iv) Facteur de conjugaison: $\gamma \leftarrow (\mathbf{g}, \mathbf{g}_p)$
 1ère itération: $\mathbf{g}_c \leftarrow \mathbf{g}_p$
 Itérations suivantes: $\mathbf{g}_c \leftarrow \mathbf{g}_p + \frac{\gamma}{\gamma_{-1}}\mathbf{g}_c$
 stocker le produit scalaire: $\gamma_{-1} \leftarrow \gamma$
- (v) Synthétique de la direction de descente: $\mathbf{l}\mathbf{g}_c \leftarrow \underline{L}\mathbf{g}_c$
- (vi) Recherche linéaire du minimum, et mise à jour $\mathbf{u} \leftarrow \mathbf{u} + \alpha\mathbf{g}_c$
 $\mathbf{r} \leftarrow \mathbf{r} - \alpha\mathbf{l}\mathbf{g}_c$
- (vii) aller à ii) ou arrêter.

On a ponctuellement besoin d'un tableau supplémentaire \mathbf{g}_p pour avoir simultanément en mémoire le gradient et le gradient preconditionné (étape (iv)). Si on dispose explicitement de l'opérateur \underline{Q} , et que son application dans un produit scalaire ne nécessite pas d'espace mémoire supplémentaire (et qu'elle est très rapide), on peut se passer de ce tableau et n'utiliser que \mathbf{g} :

- (iii) $\gamma \leftarrow (\mathbf{g}, \underline{Q}\mathbf{g})$, et preconditionnement: $\mathbf{g} \leftarrow \underline{Q}\mathbf{g}$
- (iv) Facteur de conjugaison:
 1ère itération: $\mathbf{g}_c \leftarrow \mathbf{g}$
 Itérations suivantes: $\mathbf{g}_c \leftarrow \mathbf{g} + \frac{\gamma}{\gamma_{-1}}\mathbf{g}_c$
 stocker le produit scalaire: $\gamma_{-1} \leftarrow \gamma$

3.5.3 - Poids non constants- Problèmes non quadratiques - Algorithmes IRLS

3.5.3.1 - Implémentation

Dans les cas où l'on veut minimiser une forme non quadratique, on l'approxime localement par des formes quadratiques, ce qui se traduit en général par des poids non plus constants, mais qui dépendent du modèle (directement ou par l'intermédiaire des résidus). On entre alors dans la famille des algorithmes **IRLS (Iteratively Reweighted Least-Squares)**, qui ont pour base des algorithmes de

résolution classiques à poids constants (Voir Scales and Gersztenkorn, 1987, pour une présentation claire des algorithmes IRLS).

On peut distinguer deux types d'algorithmes IRLS. Le premier (**type 1**) consiste à résoudre plusieurs fois de suite (E3.22) à poids constant (par la méthode que l'on veut, directe ou itérative), en recalculant entre chaque résolution le poids en fonction de la dernière solution connue. Si on utilise une méthode itérative, on a le choix pour chaque résolution d'utiliser comme point de départ des itérations la solution obtenue par la résolution précédente, ou de repartir d'un modèle nul. Le second (**type 2**) utilise la nature itérative d'une méthode de résolution (gradient conjugué par exemple), pour effectuer une mise à jour des poids d'itération en itération pendant la résolution (les mises à jour ne sont pas obligatoirement effectuées à chaque itération). Une fois les poids mis à jour, toutes les opérations (calcul du gradient, recherches linéaires, ...) sont effectuées à poids constants.

La preuve de la convergence donnée en annexe B suppose que chaque minimisation à poids constant est complète (on va jusqu'au minimum), ce qui correspond à un algorithme IRLS de type 1. Elle est également valable pour l'algorithme de type 2 dans la mesure où le point de départ de chaque minimisation est le point d'arrivée de la précédente.

On adapte ici le gradient conjugué pour construire un algorithme IRLS de type 2, en modifiant les poids pendant les itérations. Dans ce cas les formules de conjugaison ne sont plus exactes. Il est même plus juste de dire qu'elles n'ont plus vraiment de signification, le minimum à atteindre changeant à chaque itération. Il est donc préférable de revenir à l'esprit du gradient conjugué, à savoir que la formule de conjugaison permet de réduire une recherche bilinéaire sur la direction de descente précédente et le nouveau gradient à une recherche linéaire simple, en sachant calculer à priori la nouvelle direction de descente. On effectue ici la recherche bilinéaire exacte:

(v) Initialisations: $\mathbf{u} \leftarrow \mathbf{u}_0$ (en général $\mathbf{u}_0 = \mathbf{0}$)
 $\mathbf{r} \leftarrow \mathbf{d} - \mathbf{L}\mathbf{u}_0$

(vi) Mise à jour des poids $\tilde{\mathbf{P}}_d$ et $\tilde{\mathbf{P}}_u$ (pas nécessairement à chaque itération)

(vii) Calcul du gradient (éventuellement préconditionné): $\mathbf{g} \leftarrow \mathbf{Q}(\tilde{\mathbf{L}}^* \tilde{\mathbf{P}}_d \mathbf{r} - \tilde{\mathbf{P}}_u \mathbf{u})$

(viii) Synthétique du gradient: $\mathbf{l}\mathbf{g} \leftarrow \tilde{\mathbf{L}}\mathbf{g}$

(ix) 1ère itération: recherche linéaire le long de \mathbf{g} :
 $\mathbf{g}_c \leftarrow \alpha \mathbf{g}$
 $\mathbf{l}\mathbf{g}_c \leftarrow \alpha \mathbf{l}\mathbf{g}$

Itérations suivantes: recherche bi-linéaire dans le plan $\{\mathbf{g}, \mathbf{g}_c\}$:
 $\mathbf{g}_c \leftarrow \alpha \mathbf{g} + \beta \mathbf{g}_c$
 $\mathbf{l}\mathbf{g}_c \leftarrow \alpha \mathbf{l}\mathbf{g} + \beta \mathbf{l}\mathbf{g}_c$

(x) Mise à jour
 $\mathbf{u} \leftarrow \mathbf{u} + \mathbf{g}_c$
 $\mathbf{r} \leftarrow \mathbf{r} - \mathbf{l}\mathbf{g}_c$

(xi) aller à ii) ou arrêter.

La quantité de calculs est à peu près identique à celle du schéma classique. La recherche bilinéaire nécessite 10 produits scalaires au lieu de 4 pour la recherche linéaire (5 en incluant le calcul du facteur de conjugaison).

3.5.3.2 - Minimisation d'une forme non quadratique

La minimisation de formes pseudo-quadratiques où les poids ne sont pas constants peut résulter de la minimisation d'une fonction quelconque (pas tout à fait quand même) du modèle (voir l'annexe B). Si on veut minimiser la fonction coût:

$$\varphi(\mathbf{u}) = \sum_{i=1}^{n_d} f_i^d(r_i) + \sum_{i=1}^{n_u} f_i^u(u_i) \quad (\text{E3.27})$$

Où les $f_i(\cdot)$ sont des fonctions scalaires qui répondent aux hypothèses (EA.22):

(i) f_i de classe C1 (ii) $f_i''(0)$ existe et est finie (iii) f_i strictement convexe (iv) f_i paire (v) $p_i(r)$ décroissante pour ($r \geq 0$)	(E3.28)
---	---------

Avec

$$\begin{cases} p_i(r \neq 0) = \frac{f_i'(r)}{r} \\ p_i(0) = f_i''(0) \end{cases} \quad (\text{E3.29})$$

n_d est la dimension de \mathbf{d} , n_u est la dimension de \mathbf{u} . On pose:

$$\tilde{\mathbf{L}}\mathbf{u} = \begin{bmatrix} \tilde{\mathbf{L}} \\ \tilde{\mathbf{I}} \end{bmatrix} \mathbf{u} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix} = d \quad r = d - \tilde{\mathbf{L}}\mathbf{u} = \begin{bmatrix} \mathbf{r} \\ -\mathbf{u} \end{bmatrix} \quad (\text{E3.30})$$

Alors:

$$\varphi(\mathbf{u}) = \sum_{i=1}^{n_d+n_u} f_i(r_i) \quad \begin{cases} f_i = f_i^d & i \leq n_d \\ f_i = f_{i-n_d}^u & i > n_d \end{cases} \quad (\text{E3.31})$$

On s'est donc ramené ainsi à la forme décrite dans l'annexe B.

La minimisation de (E3.31) se fait alors par des minimisations successives à poids constant des formes quadratiques:

$$\psi(\mathbf{u}) = \frac{1}{2}(r, \tilde{\mathbf{P}}r) \quad (\text{E3.32})$$

En choisissant pour $\tilde{\mathbf{P}}$, poids diagonal (équ. (EA.23)):

$$(\tilde{\mathbf{P}})_{ii} = \frac{f_i'(r_i)}{r_i} = p_i(r_i) \quad (\text{E3.33})$$

Ce poids est décomposable en poids sur les données et sur le modèle:

$$\tilde{P} = \begin{bmatrix} \tilde{\mathbf{P}}_d & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{P}}_u \end{bmatrix} \quad \begin{aligned} (\tilde{\mathbf{P}}_d)_{ii} &= p_i(r_i) = p_i^d(r_i) \\ (\tilde{\mathbf{P}}_u)_{ii} &= p_{n_d+i}(r_{n_d+i}) = p_i^u(u_i) \end{aligned} \quad (\text{E3.34})$$

et (E3.32) est alors identique à (E3.22):

$$\begin{aligned} \psi(\mathbf{u}) &= \frac{1}{2}(r, Pr) = \frac{1}{2} \left(\begin{bmatrix} \mathbf{r} \\ -\mathbf{u} \end{bmatrix}, \begin{bmatrix} \tilde{\mathbf{P}}_d \mathbf{r} \\ \tilde{\mathbf{P}}_u (-\mathbf{u}) \end{bmatrix} \right) \\ &= \frac{1}{2}(\mathbf{r}, \tilde{\mathbf{P}}_d \mathbf{r}) + \frac{1}{2}(\mathbf{u}, \tilde{\mathbf{P}}_u \mathbf{u}) \end{aligned}$$

3.5.3.3 - Poids non constant donnés à priori

Si on se donne à priori les fonctions poids p_i , alors pour que les fonctions $f_i(R) = \int_0^R r p_i(r) dr$ vérifient les conditions (E3.28), il faut que:

(i) p_i continue (ii) p_i paire (iii) $p_i(r)$ décroissante pour $(r \geq 0)$ (iv) $r p_i(r)$ strictement croissante pour $(r \geq 0)$	(E3.35)
---	---------

3.6 - Inversion en norme L2: propriétés

3.6.1 - Régularisation - facteur d'amortissement

Lorsque l'on résout le problème en norme L2, on cherche à minimiser la fonction coût:

$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}\|^2 + \frac{\varepsilon}{2} \|\mathbf{u}\|^2 \quad (\text{E3.36})$$

conduisant à la solution:

$$\mathbf{u} = (\tilde{\mathbf{L}}^* \tilde{\mathbf{L}} + \varepsilon \mathbf{I})^{-1} \tilde{\mathbf{L}}^* \mathbf{d} \quad (\text{E3.37})$$

Quelle est la signification du facteur d'amortissement ε et comment le déterminer ? On lui assigne deux rôles: régulariser l'inversion, qui sinon pourrait avoir une infinité de solutions, et prendre en compte la présence de bruit sur les données.

Pour se convaincre que le problème de la minimisation de l'énergie des résidus est mal posé, il suffit d'essayer plusieurs types de régularisations (fig. 3.1): en 3.1a on se donne un modèle contenant 5 évènements paraboliques. La modélisation correspondante est en 3.1b.

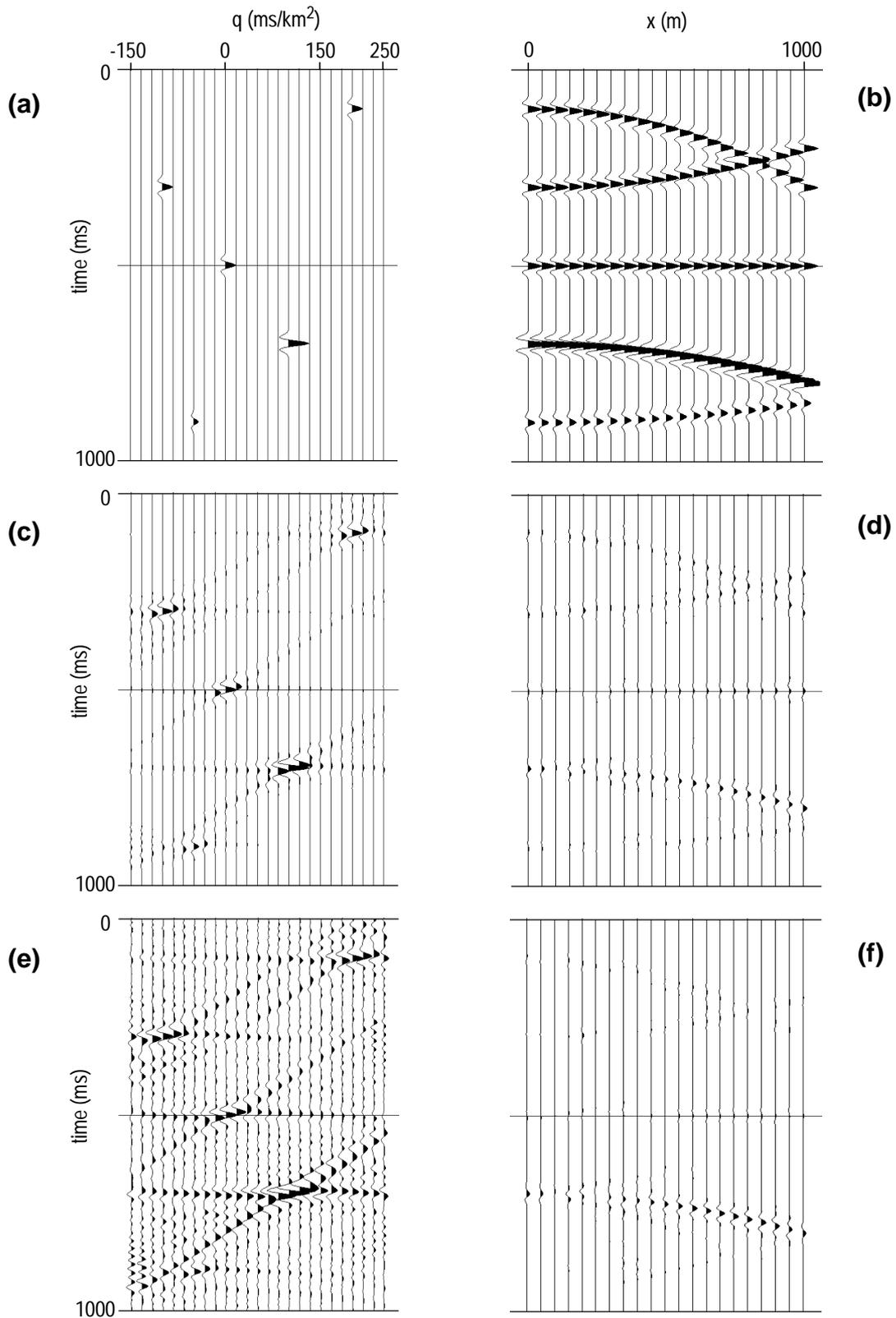


Figure 3.1 : Influence de la régularisation sur la Décomposition de Radon Parabolique. (a) modèle contenant 5 évènements. (b) synthétique obtenue par modélisation de (a). (c) PRD de (b) en norme L2. (d) les résidus correspondant. (e) PRD en norme L4 de (b). (f) les résidus correspondant. (gain (d) et (f))=2 X gain(b)).

En 3.1c se trouve la PRD en norme L2 de 3.1b, obtenue par minimisation de (E3.36) (résidus en 3.1d). En 3.1e se trouve la PRD en norme L4 sur le modèle, obtenue par minimisation de:

$$\varphi(\mathbf{u}) = \frac{1}{2}\|\mathbf{d} - \mathbf{L}\mathbf{u}\|_{L2}^2 + \frac{\varepsilon}{2}\|\mathbf{u}\|_{L4}^4 \quad (\text{résidus en 3.1f}).$$

Les trois modèles fig. 3.1(a)(c)(e) sont tous des bonnes solutions à la décomposition de Radon parabolique de l'image 3.1b, en ce sens que les résidus sont tous faibles (énergie résiduelle inférieure à 0.5%; 3.1a, le modèle de départ, est même une solution à résidus nuls). Les modèles ont néanmoins des aspects très différents, imposés par le terme de régularisation.

Le deuxième problème est la détermination du facteur d'amortissement. Théoriquement, il est lié au niveau de bruit présent sur les données et à l'amplitude des valeurs attendues sur le modèle. Mais en général, ce paramètre ne fait pas l'objet d'une détermination rigoureuse. La plupart du temps, on choisit une fraction des élément diagonaux de $\mathbf{L}^*\mathbf{L}$ (typiquement 1%). Gulunay (1990), pour les LRD et PRD, recommande d'utiliser un facteur d'amortissement très faible (jusqu'à 0.01%), ce qui améliore la focalisation des évènements sur le modèle (plus le facteur d'amortissement augmente, plus la solution se rapproche de la simple application de \mathbf{L}^* (équ. (E3.37))).

3.6.1.1 - Le facteur d'amortissement du point de vue bayésien

Du point de vue Bayésien, la valeur à attribuer au facteur d'amortissement n'est pas subjective. La résolution en norme L2 revient à considérer des pdf gaussiennes (équ. (E3.15)), centrées sur zéro, avec des matrices de covariances proportionnelles à l'identité. Le bruit est donc supposé à moyenne nulle et de variance uniforme σ_n^2 , de même que le modèle de variance uniforme σ_u^2 :

$$\mathbf{C}_D = \mathbf{P}_D^{-1} = \sigma_n^2 \mathbf{I}_D \quad \mathbf{C}_U = \mathbf{P}_U^{-1} = \sigma_u^2 \mathbf{I}_U \quad (\text{E3.38})$$

On doit donc minimiser (équ. (E3.2)) la quantité:

$$\varphi(\mathbf{u}) = \frac{1}{2\sigma_n^2}\|\mathbf{d} - \mathbf{L}\mathbf{u}\|^2 + \frac{1}{2\sigma_u^2}\|\mathbf{u}\|^2 \quad (\text{E3.39})$$

Le facteur d'amortissement devrait donc être déterminé d'après les valeurs de ces variances:

$$\varepsilon = \frac{\sigma_n^2}{\sigma_u^2} \quad (\text{E3.40})$$

Si il est possible d'estimer la variance du bruit (il existe plusieurs méthodes statistiques pour estimer le niveau de bruit sur un ensemble de traces), fixer à priori la variance du modèle pose un réel problème. En effet, comment, avant d'avoir calculé ce modèle, est-il possible d'en connaître les amplitudes?

Une des solutions est de tirer parti de la méthode de résolution. Une méthode itérative, telle que le gradient conjugué, permet d'avoir, à chaque itération, une approximation de la solution finale. Il est donc possible d'obtenir à chaque itération une nouvelle estimation de la variance du modèle, et donc du facteur d'amortissement. On estime la variance du modèle simplement par une moyenne quadratique des échantillons du modèle:

$$\sigma_u^2 \leftarrow (\|\mathbf{u}\|^2 / n_u) \quad (\text{E3.41})$$

Le point de départ des itérations du gradient est presque toujours le modèle nul ($\mathbf{u}_0=0$). La variance est donc certainement sous-estimée pendant les premières itérations. On peut 'estimer cette sous-estimation' en comparant la variance des résidus à la variance préalablement estimée (ou fixée arbitrairement) du bruit: plus le rapport $(\|\mathbf{r}\|^2/n_d)/\sigma_n^2$ est proche de 1, plus on est proche de la solution finale. On utilise donc ce rapport pour corriger l'estimation de la variance du modèle:

$$\sigma_u^2 \leftarrow (\|\mathbf{u}\|^2/n_u) \frac{(\|\mathbf{r}\|^2/n_d)}{\sigma_n^2} \tag{E3.42}$$

On suit donc le processus suivant:

(i) Estimer ou fixer arbitrairement la variance du bruit σ_n^2

$\mathbf{r} \leftarrow \mathbf{d}$

(ii) Initialisations: $\mathbf{u} \leftarrow \mathbf{0}$
 $\varepsilon \leftarrow 0$

(iii) calculer \mathbf{u} et \mathbf{r} par une itération de gradient conjugué sur la forme quadratique:

$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|^2 + \frac{\varepsilon}{2} \|\mathbf{u}\|^2$$

(iv) Estimer un nouveau facteur d'amortissement par la formule:

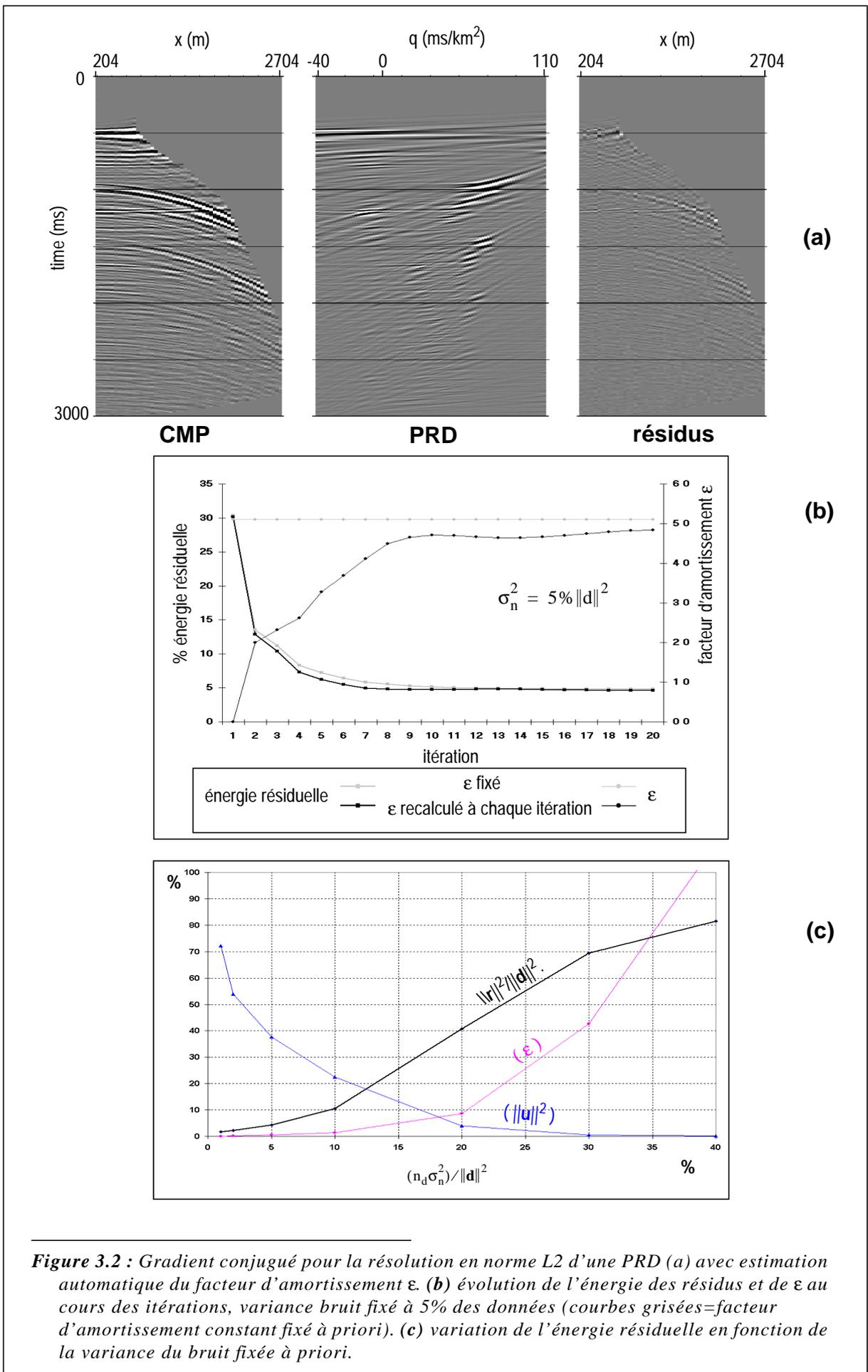
$$\varepsilon \leftarrow \frac{\sigma_n^2}{\sigma_u^2} = \frac{\sigma_n^2}{\|\mathbf{u}\|^2/n_u} \frac{\sigma_n^2}{\|\mathbf{r}\|^2/n_d}$$

(v) Aller en (iii) ou arrêter.

On définit ainsi une sorte d'algorithme IRLS, dans lequel le poids sur le modèle est uniforme mais varie d'itération en itération. Ce poids ne vérifie pas les hypothèses (E3.35) pour assurer la convergence, en particulier la décroissance par rapport au modèle (E3.35)(iii), à cause de la dépendance par rapport aux résidus. Il faut en outre introduire dans le calcul de ε des bornes inférieures pour $\|\mathbf{u}\|^2$ (sa valeur à la première itération par exemple) et pour $\|\mathbf{r}\|^2$ (une fraction des résidus visés ou estimés par exemple) pour respecter la continuité (E3.35)(i). Néanmoins les conditions (E3.35) sont seulement suffisante, pas nécessaires, et dans la pratique on constate que le processus converge et que la valeur du facteur d'amortissement se stabilise au bout de quelques itérations (figure 3.2b).

On vérifie également sur le graphique 3.2c que l'énergie résiduelle de la solution atteinte et la variance du bruit fixée a priori sont liées linéairement (elles devraient être égales, mais on constate ici l'existence d'un facteur multiplicatif). Ainsi, en fixant a priori la variance du bruit, on ne trouve dans la solution (donc dans les résidus) que ce que l'on a décidé d'y mettre, et fixer un facteur d'amortissement a priori revient à fixer une variance de bruit a priori. Il semble donc important de pouvoir bien estimer le niveau de bruit sur les données avant d'effectuer l'inversion.

De même que l'on utilise le modèle au cours des itérations pour avoir une estimation de la variance finale du modèle, on pourrait envisager d'estimer la variance du bruit d'après les résidus, soit au cours des itérations, soit à la fin de l'inversion pour refaire une inversion avec la nouvelle valeur. Malheureusement ce processus est totalement instable, comme l'a expérimenté Claerbout (1992, p. 91): suivant les valeurs initiales, le système se dirige soit vers une solution nulle, soit vers des résidus nuls.



3.6.2 - Influence du facteur d'amortissement sur l'atténuation du bruit

Un des rôles du facteur d'amortissement est de tenir compte de la présence de bruit sur les données. Il s'apparente donc à un facteur de blanchiment. On souhaite ne pas modéliser le bruit, afin de ne conserver que le signal 'utile' dans la reconstruction. Il reste à définir ici ce que l'on considère comme du bruit.

Dans le problème de la décomposition de Radon, on appelle généralement bruit tout ce qui ne rentre pas dans le modèle choisi (linéaire, parabolique, ...). Le bruit aléatoire en fait partie bien sûr, de même par exemple que les arrivées directes (linéaires) lors de la décomposition hyperbolique d'un CMP. Dans cette optique, un évènement linéaire d'amplitude non constante de trace à trace contient, pour une décomposition linéaire, du signal (la partie constante) et du bruit (le reste).

Dans la pratique, on constate que ce que l'on considérerait a priori comme du bruit est souvent parfaitement modélisable: on peut sans problème effectuer la Décomposition de Radon Linéaire d'un point de tir, composé d'hyperboles, et reconstruire ensuite parfaitement les données de départ, les hyperboles étant parfaitement modélisables par une superposition de droites.

Reprenons la collection CMP de la figure 3.1b, contenant 5 évènements paraboliques -dont un plat-, auquel nous ajoutons une certaine quantité de bruit gaussien (fig. 3.3), et effectuons des LRD en norme L2 avec plusieurs valeurs de facteur d'amortissement.

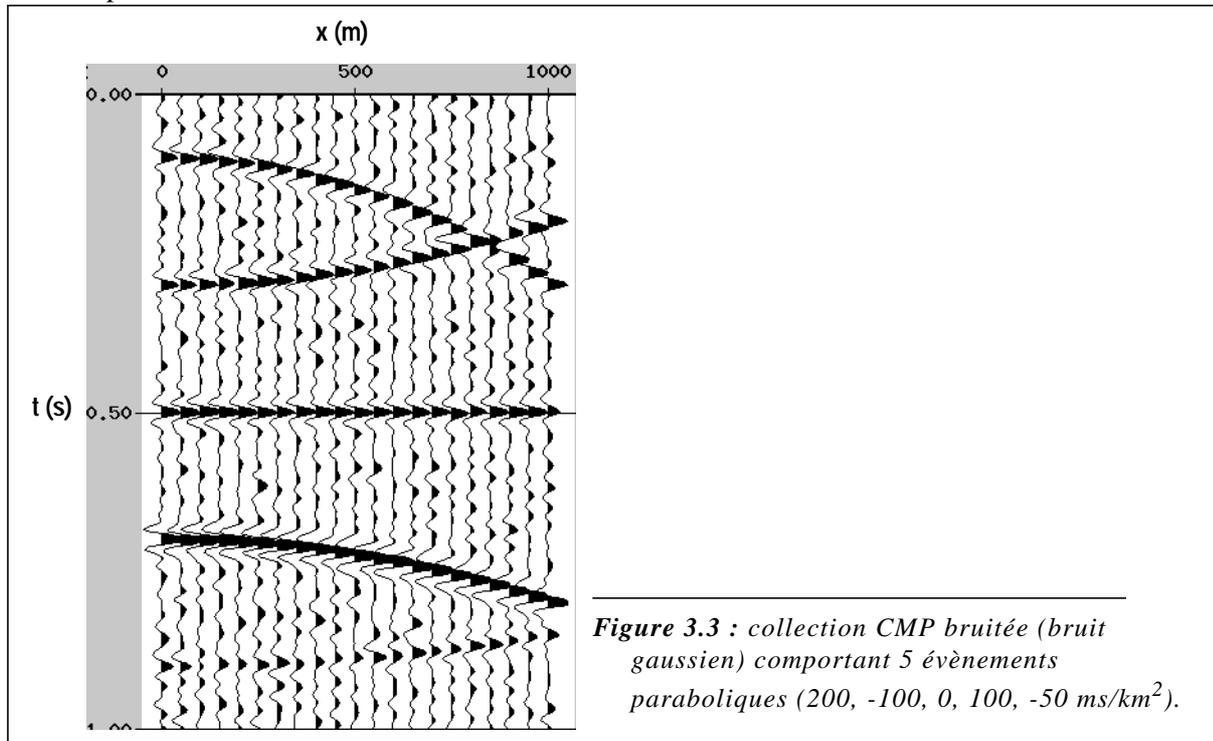
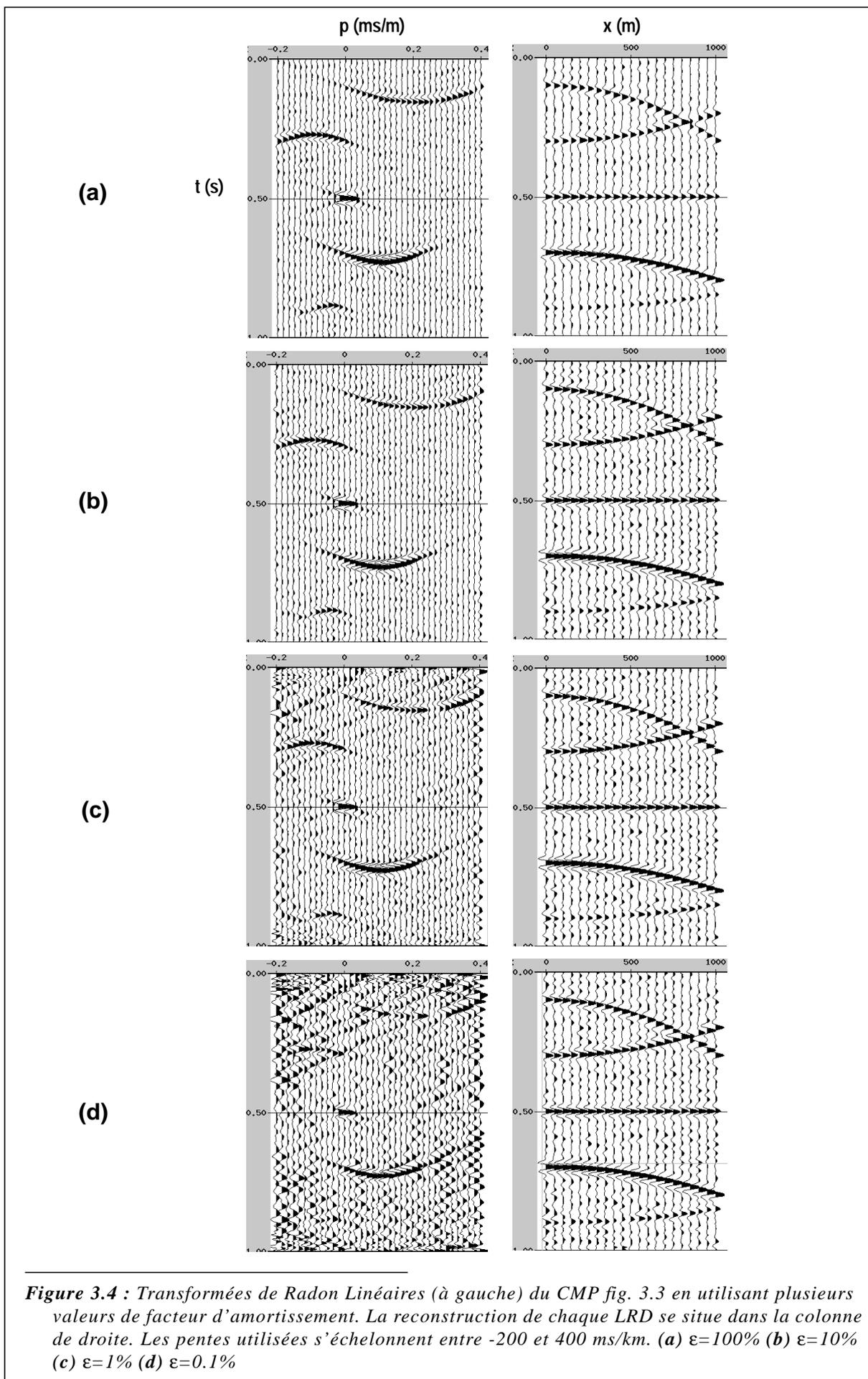


Figure 3.3 : collection CMP bruitée (bruit gaussien) comportant 5 évènements paraboliques (200, -100, 0, 100, -50 ms/km²).

Un facteur de blanchiment très faible (0.1%) conduit à un bruitage excessif de la solution (fig. 3.4d), en laissant apparemment peu de bruit dans les résidus. Le bruit qui 'explose' sur le modèle appartient au noyau de \mathbf{L} et n'est pas forcément lié au bruit présent sur les données. Le facteur d'amortissement permet en principe d'imposer une composante nulle dans le noyau (voir la figure 1.6).

Lorsque le facteur d'amortissement passe à 1% puis à 10% (fig. 3.4c et 3.4b), on réduit notablement le niveau de bruit sur le modèle, mais pas sur les reconstructions: le bruit est modélisé aussi bien que les évènements cohérents. Pour un facteur d'amortissement de 100%, on note bien une certaine atténuation du bruit, mais également du signal cohérent, de sorte que le rapport S/B ne semble pas amélioré (ou très peu).

En résumé il ne semble pas possible d'améliorer notablement le rapport S/B par une augmentation du facteur d'amortissement. Seul le bruit perpendiculaire à l'image de \mathbf{L} peut être filtré efficacement par la Transformée de Radon. On a ici la même difficulté à discriminer le signal cohérent du bruit que celle



que l'on aurait à vouloir discriminer un bruit aléatoire d'une sinusoïde pure par une transformée de Fourier: les deux sont aussi bien représentés dans le domaine des fréquences.

En norme L2, le facteur d'amortissement permet de régulariser l'inversion, mais pas d'améliorer le rapport Signal/Bruit dans l'espace des données.

3.6.3 - Considérations fréquentielles pour la LRD

On choisit un modèle \mathbf{u}_0 (fig. 3.5a) contenant un spike (=dirac numérique) en $p=0$, définissant donc un évènement horizontal pur portant un signal (fig. 3.5c). La représentation fréquentielle de ce modèle (fig. 3.5b) montre un spectre constant sur la trace $p=0$.

En fig. 3.5e-f se trouvent \mathbf{u}_S (qui est le stack de \mathbf{d} , ou encore la convolution de \mathbf{u}_0 par σ), et sa représentation fréquentielle:

$$\hat{\mathbf{u}}_S(\omega, p) = \mathbf{L}^* \hat{\mathbf{d}}(\omega, p) = \hat{\mathbf{u}}_0(\omega, p) *_{p} \sigma_{\text{lin}}(\omega, p) = \delta(p) *_{p} \sigma_{\text{lin}}(\omega, p) = \sigma_{\text{lin}}(\omega, p)$$

On trouve donc sur chaque ligne de $\hat{\mathbf{u}}_S$ l'opérateur de convolution pour chaque fréquence: l'étalement du spike (la largeur de l'opérateur de convolution) est inversement proportionnel à la fréquence. La trace $p=0$ de \mathbf{u}_S contient le même signal que le signal original, stacké avec la pente exacte de l'évènement.

$\hat{\mathbf{u}}_2$, qui est la déconvolution de $\hat{\mathbf{u}}_S$ par rapport à p , est en fig. 3.5g-h: l'énergie étalée est partiellement reconcentrée en $p=0$, où idéalement on aurait voulu retrouver un spike. Mais il reste néanmoins un certain étalement inversement proportionnel à la fréquence. Par conséquent, la trace centrale $p=0$ présente un contenu plus HF que le modèle de départ, les BF se retrouvant (relativement) prépondérantes sur les traces p voisines.

En fig. 3.5d on vérifie que la modélisation de \mathbf{u}_2 est égale aux données de départ (fig. 3.5c), c'est à dire que \mathbf{u}_2 est bien une décomposition correcte de ces données.

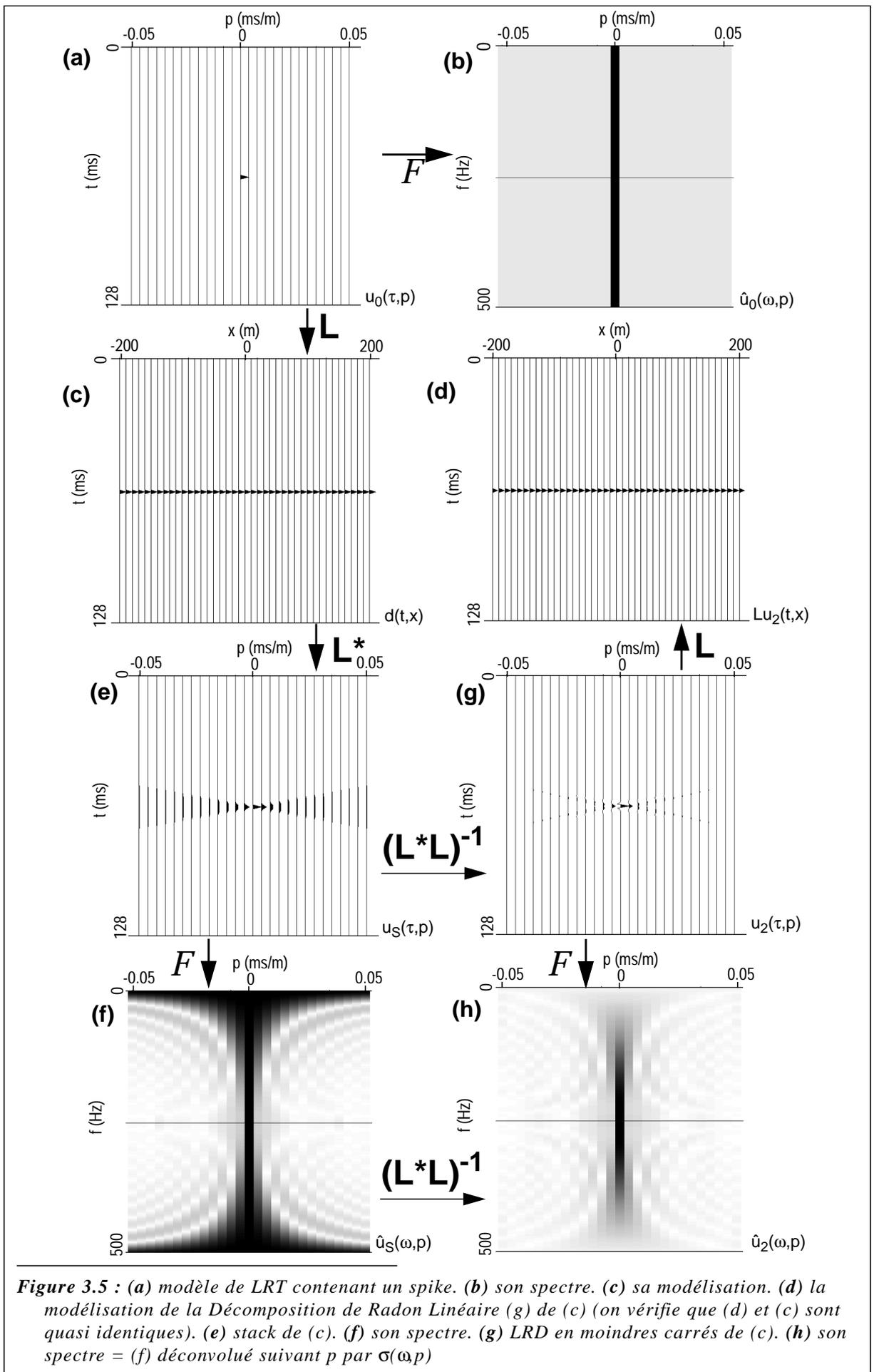
En conclusion, la LRD en moindres carrés d'un évènement pur (parfaitement linéaire, signal constant) donne une image qui n'est pas ponctuelle, mais qui possède une certaine extension suivant p . De plus les fréquences contenues dans le signal se répartissent différemment, les HF étant plus concentrées au centre de l'image. Par conséquent, le signal extrait sur une seule trace de la solution n'est pas identique au signal d'origine (même sur la trace p correspondant à la pente exacte de l'évènement): il faut considérer l'image dans son ensemble pour reconstituer celui-ci.

Ce résultat est cohérent avec les résultats du paragraphe 1.9.5.1. En effet, le filtre rho à appliquer à l'image stack pour approximer l'inversion est en $|\omega|$, ce qui relève les HF sur la trace centrale. On pourrait constater le même comportement avec les transformées paraboliques ou hyperboliques, avec des amplifications plus ou moins importantes des HF (en parabolique, le filtre rho est en $\sqrt{|\omega|}$).

3.7 - Influence de la couverture

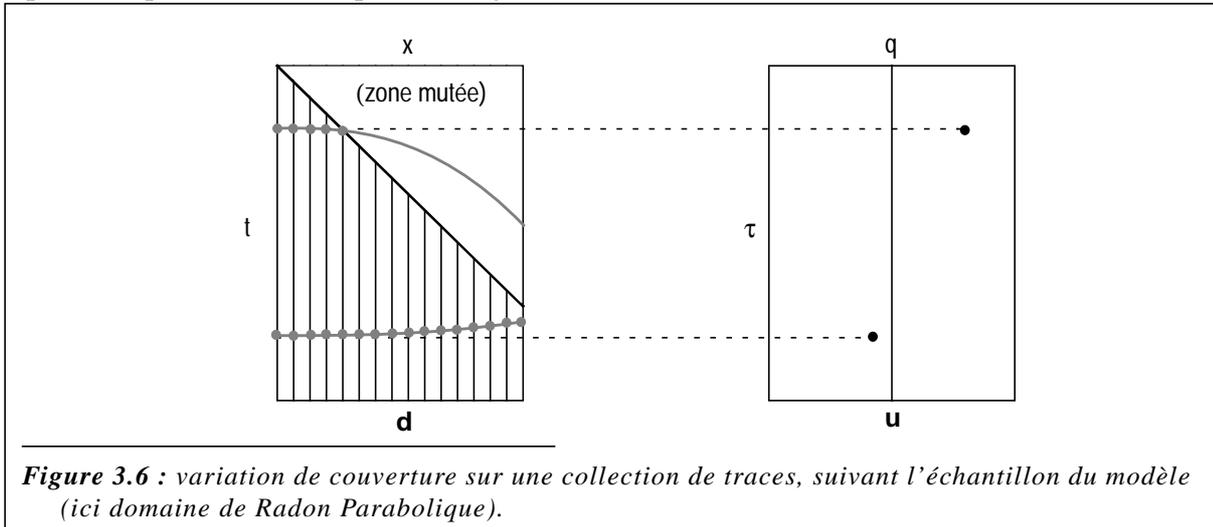
3.7.1 - Pondération du modèle

Les points de tir, et donc les collections CMP, sont souvent 'mutés' pour éliminer les arrivées directes et les réfractions. De plus, les corrections NMO sur les CMP introduisent un stretch du signal, que l'on limite par un autre mute dans les zones où il est trop important. Ces mutes ont pour conséquence



d'introduire des variations de couverture dans une collection de traces donnée (voir par exemple la collection CMP fig. 3.2).

Lorsque l'on effectue les Décompositions de Radon en domaine de Fourier, on ne peut pas tenir compte des zones mutées, qui sont assimilées à des échantillons nuls. Le domaine temps permet de ce point de vue plus de souplesse car les zones mutées peuvent réellement être traitées comme une absence de données. Néanmoins ces variations de couverture dans une collection de traces risquent de conduire à un traitement différent des zone à faible et à forte couverture. En effet, à un point donné du modèle correspond une courbe dans l'espace des données, et le nombre d'échantillons sur lesquels elle est représentée peut avoir de l'importance (fig. 3.6).



La résolution classique en norme L2 fait intervenir un terme de minimisation de l'énergie du résidu et un terme de minimisation de l'énergie du modèle: $\varphi(\mathbf{u}) = \frac{1}{2}\|\mathbf{d} - \underline{\mathbf{L}}\mathbf{u}\|^2 + \frac{\varepsilon}{2}\|\mathbf{u}\|^2$

Chaque échantillon du modèle est mis en balance avec une somme de carrés d'échantillons du résidu (situés sur la trajectoire de la courbe correspondante), dont le nombre de terme est égal à la couverture le long de la trajectoire considérée. On aurait donc envie de compenser cet effet en appliquant dans l'espace des modèles un poids égal à cette couverture: $\underline{\mathbf{C}}$ est un poids diagonal qui contient la couverture associée à chaque échantillon du modèle, et on minimise:

$$\varphi_c(\mathbf{u}) = \frac{1}{2}\|\mathbf{d} - \underline{\mathbf{L}}\mathbf{u}\|^2 + \frac{\varepsilon}{2}(\mathbf{u}, \underline{\mathbf{C}}\mathbf{u}) \tag{E3.43}$$

Un critère pour juger de la nécessité d'appliquer ce poids sur le modèle est de regarder les résultats sur des évènements de mêmes amplitudes mais de couvertures différentes (fig. 3.7). On voudrait que les résidus après décomposition de Radon soient d'amplitudes comparables pour chaque évènement.

On effectue deux LRD en norme L2, l'une sans pondération par la couverture, l'autre avec pondération (fig. 3.8): les résidus sans pondération sont uniformes, alors qu'avec pondération ils sont d'autant plus faibles que la couverture est faible. Concernant les modèles, il est difficile de voir une différence à l'oeil nu. Le raisonnement sur le nombre d'échantillons en jeu dans chacun des deux termes de la minimisation est donc incomplet: on oublie qu'en norme L2, un évènement est représenté dans l'espace modèle non pas sur un seul mais sur plusieurs échantillons (il s'étale), et cela d'autant plus que la couverture est faible. On vérifie d'ailleurs aisément que ce raisonnement est valable si on ne prend qu'une seule trace (p=0) dans le modèle, obligeant ainsi les évènements à se former sur un seul échantillon (fig 3.8): sans pondération, les résidus sont plus forts là où la couverture est faible.

Le problème se pose différemment avec une norme L1 sur le modèle (minimisation de la norme L1 du modèle au lieu de la norme L2, voir chap. 4): celle-ci tend en effet à concentrer la représentation des

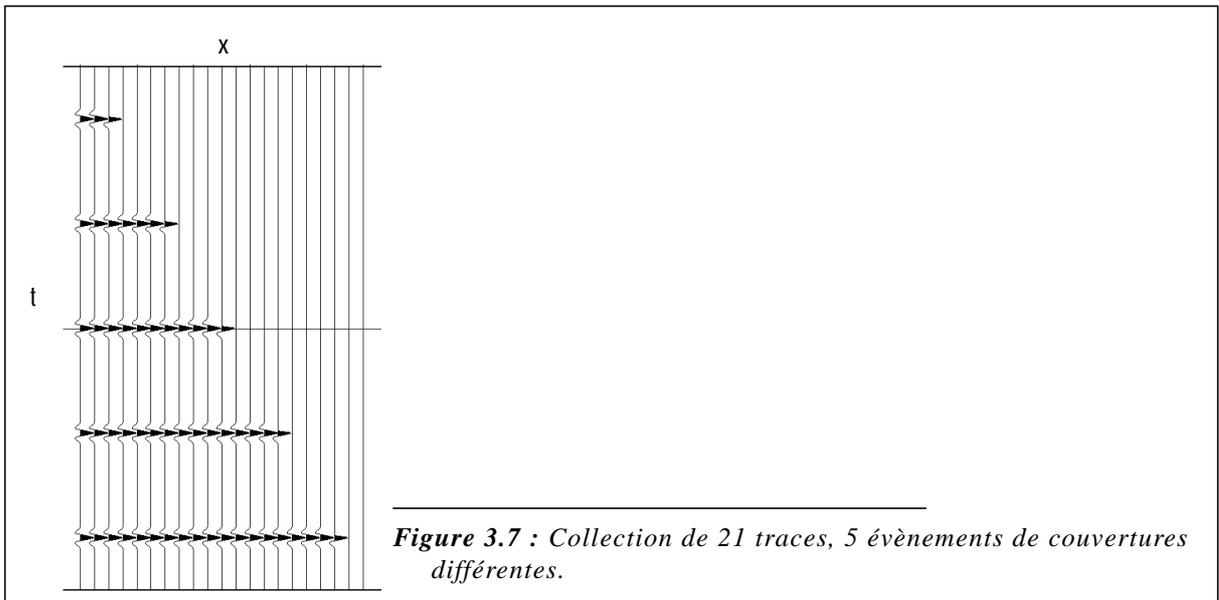


Figure 3.7 : Collection de 21 traces, 5 évènements de couvertures différentes.

évènements sur un nombre réduit d'échantillons. On effectue ici deux LRD en normes L1 (fig. 3.8), et on constate qu'il est préférable de pondérer par la couverture pour que les résidus soient uniformes.

En généralisant, si une norme L_p (p étant compris entre 1. et 2.) est appliquée sur le modèle, il est souhaitable de pondérer le modèle par la couverture élevée à la puissance $(2-p)$.

3.7.2 - Préconditionnement du gradient

La pondération du modèle pour compenser les variations de couverture étant résolu, il reste à optimiser le calcul du gradient.

La fonction coût à minimiser, d'après les résultats du paragraphe précédent, est:

$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|^2 + \frac{\varepsilon}{2} (\mathbf{u}, \mathbf{P}_p \mathbf{C}^{2-p} \mathbf{u}) \quad (\text{E3.44})$$

\mathbf{P}_p est le poids permettant de simuler la norme L_p (chap. 4). Le gradient correspondant est:

$$\mathbf{g}(\mathbf{u}) = -\mathbf{L}^*(\mathbf{d} - \mathbf{L}\mathbf{u}) + \varepsilon \mathbf{P}_p \mathbf{C}^{2-p} \mathbf{u} \quad (\text{E3.45})$$

Les études sur la norme L_p indiquent qu'il faut preconditionner le gradient par \mathbf{P}_p^{-1} . Faut-il de la même façon le preconditionner par un terme liée à la couverture (\mathbf{C}^{p-2} par exemple) ?

Les courbes de convergence du gradient conjugué (fig. 3.10) pour la norme L2 et la norme L1, avec et sans preconditionnement (par \mathbf{C}^{-1}), montrent qu'en norme L2 le preconditionnement détériore la convergence, notamment à la première itération. En norme L1 par contre, la convergence est bien meilleure avec preconditionnement (deux fois plus rapide sur cet exemple).

En conclusion, la norme L1 réclame, outre la pondération du modèle par la couverture, le preconditionnement du gradient par l'inverse de la couverture. On généraliser ce résultat expérimental à toute norme L_p :

En appliquant une norme L_p sur le modèle, il faut:

- pondérer le modèle par \mathbf{C}^{2-p}
- preconditionner le gradient par \mathbf{C}^{p-2}

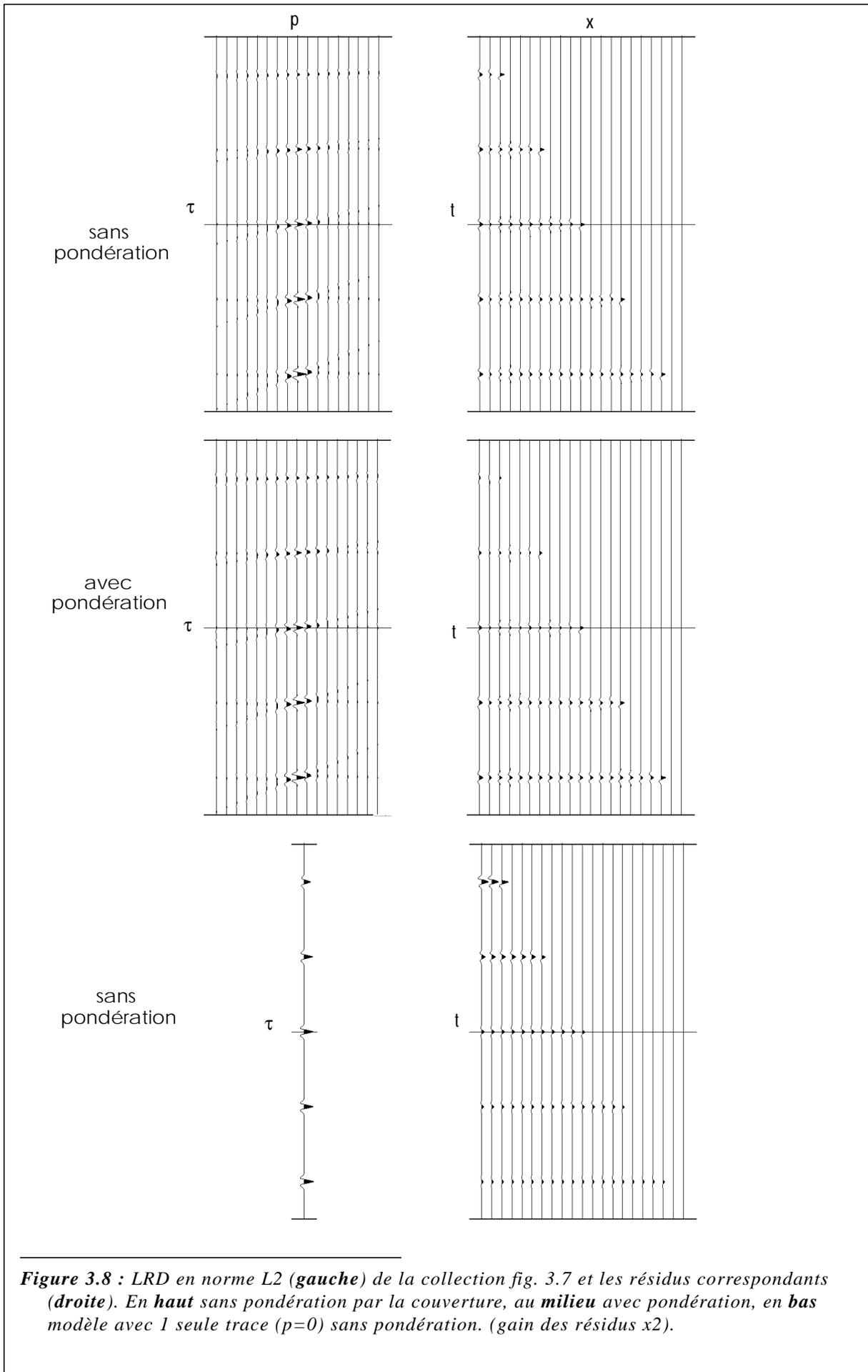


Figure 3.8 : LRD en norme L2 (gauche) de la collection fig. 3.7 et les résidus correspondants (droite). En haut sans pondération par la couverture, au milieu avec pondération, en bas modèle avec 1 seule trace ($p=0$) sans pondération. (gain des résidus $\times 2$).

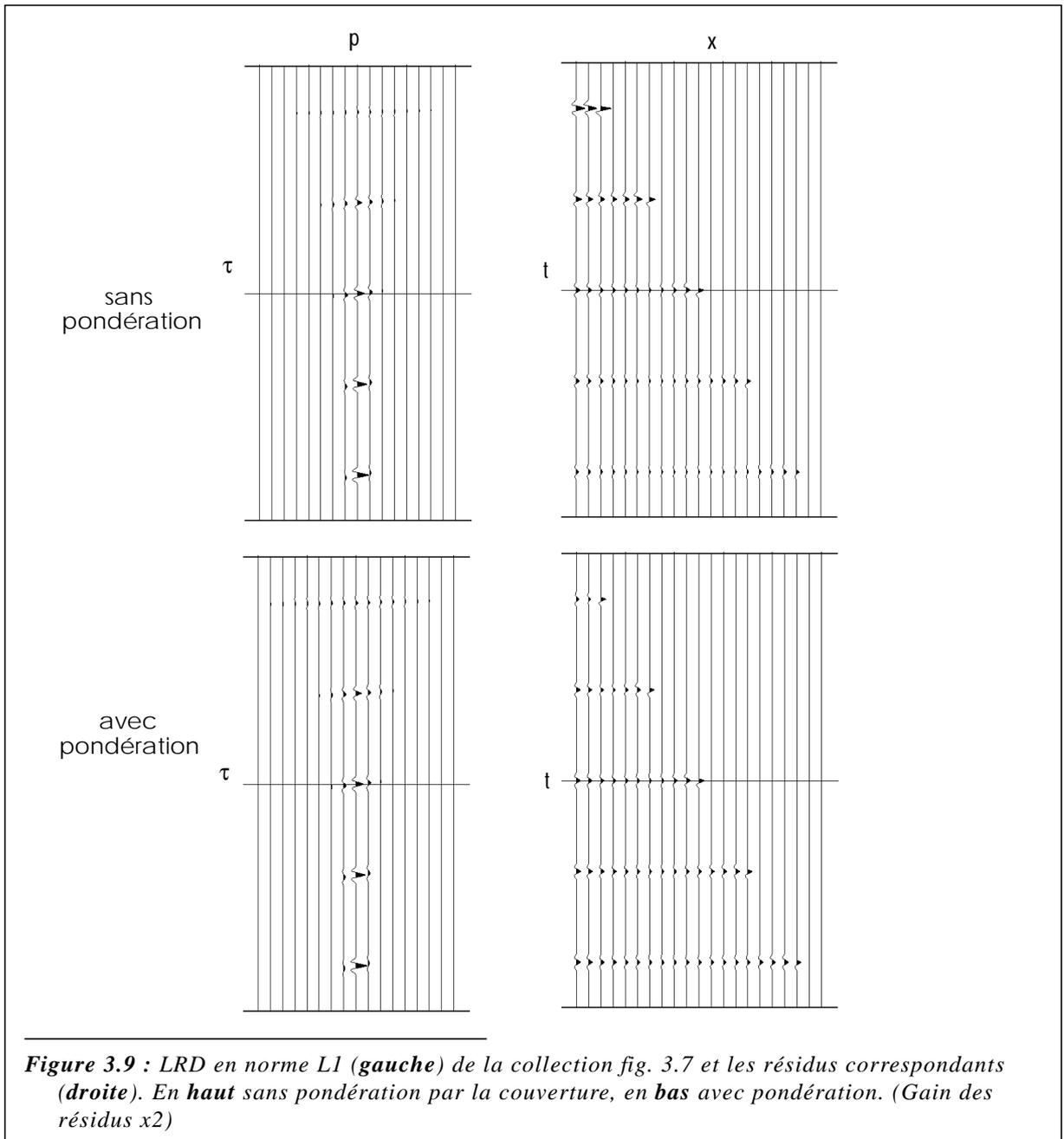


Figure 3.9 : LRD en norme L1 (gauche) de la collection fig. 3.7 et les résidus correspondants (droite). En haut sans pondération par la couverture, en bas avec pondération. (Gain des résidus $\times 2$)

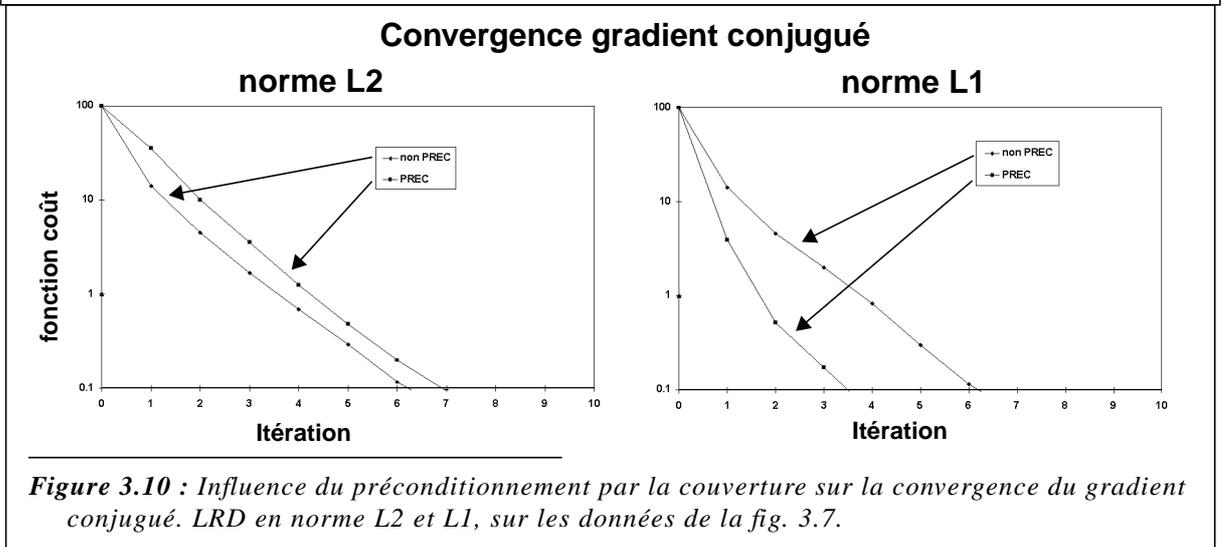


Figure 3.10 : Influence du préconditionnement par la couverture sur la convergence du gradient conjugué. LRD en norme L2 et L1, sur les données de la fig. 3.7.

3.8 - Préconditionnement par un filtre rho

3.8.1 - Filtrés simples

Dans le cas théorique de domaines continus et d'ouvertures infinies, la LRD en norme L2 se résume à l'application de l'opérateur adjoint (qui est l'opérateur de stack), suivi de l'application d'un 'filtre rho', agissant uniquement sur la variable temps. Le spectre de ce filtre est $|\omega|$ (équ. (E1.29) et paragr. 1.9.5.1)

En pratique (ouvertures finies et variables discrètes), l'application du filtre rho après l'opérateur adjoint ne donne pas exactement l'inverse, mais en réalise une bonne approximation (qui est souvent suffisante et utilisée à la place de l'inverse exact). Or dans la résolution par gradient conjugué, le gradient à chaque itération est obtenu par application de l'opérateur adjoint sur les résidus (donc sur les données à la première itération), d'où l'idée d'utiliser le filtre rho comme opérateur de preconditionnement.

On minimise la quantité: $\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d} - \mathbf{L}\mathbf{u}, \mathbf{d} - \mathbf{L}\mathbf{u})^2 + \frac{\varepsilon}{2}(\mathbf{u}, \mathbf{u})$

le gradient est: $\mathbf{g}(\mathbf{u}) = -\mathbf{L}^*(\mathbf{d} - \mathbf{L}\mathbf{u}) + \varepsilon \mathbf{u}$

Soit \mathbf{R} l'opérateur du filtre rho. Pour pouvoir utiliser le gradient preconditionné $\mathbf{R}\mathbf{g}(\mathbf{u})$, il faut vérifier que \mathbf{R} est défini, positif et symétrique:

\mathbf{R} correspond à la convolution par un filtre $\rho(t)$, de spectre $\hat{\rho}(\omega) = |\omega|$.

Le spectre est réel, donc ρ est pair: $\rho(-t) = \rho(t)$

On suppose qu'un modèle \mathbf{u} ne comporte qu'une seule trace $u(t)$. Si il y a plusieurs traces, il suffit d'ajouter des sommes à tous les calculs suivants:

Le produit scalaire $(\mathbf{u}, \mathbf{R}\mathbf{v})$ peut s'écrire:

$$(\mathbf{u}, \mathbf{R}\mathbf{v}) = \int_t u(t)(\rho * v)(t) dt$$

\mathbf{R} est défini:

si $\mathbf{u}=\mathbf{0}$ alors $(\mathbf{u}, \mathbf{R}\mathbf{u}) = 0$

si $(\mathbf{u}, \mathbf{R}\mathbf{u}) = 0$ alors:

$$\begin{aligned} \int_t u(t)(\rho * u)(t) dt &= \int_{\omega} \hat{u}(\omega) \overline{(\hat{\rho} * \hat{u})(\omega)} d\omega \\ &= \int_{\omega} \hat{u}(\omega) \overline{\hat{u}(\omega)} \hat{\rho}(\omega) d\omega = \int_{\omega} |\hat{u}(\omega)|^2 \hat{\rho}(\omega) d\omega = 0 \end{aligned}$$

$\hat{\rho}(\omega \neq 0) > 0$ entraîne que $|\hat{u}(\omega \neq 0)| = 0$. En supposant que $|\hat{u}(0)| = 0$, on a $\mathbf{u}=\mathbf{0}$.

\mathbf{R} est positif:

si $\mathbf{u} \neq \mathbf{0}$ (avec $|\hat{u}(0)| = 0$), les équations précédentes montrent également que $(\mathbf{u}, \mathbf{R}\mathbf{u}) > 0$

\mathbf{R} est symétrique:

$$\begin{aligned}
 (\mathbf{u}, \mathbf{R}\mathbf{v}) &= \int_t u(t)(\rho * v)(t)dt = \int_t u(t) \int_{t'} v(t')\rho(t-t')dt'dt = \int_t \int_{t'} u(t)v(t')\rho(t-t')dt'dt \\
 &= \int_t \int_{t'} u(t)v(t')\rho(-t+t')dt'dt = \int_{t'} v(t')(\rho * u)(t')dt' \\
 &= (\mathbf{R}\mathbf{u}, \mathbf{v})
 \end{aligned}$$

\mathbf{R} est donc utilisable pour préconditionner le gradient, à condition de travailler avec des signaux à moyenne nulle (pas de composante continue). Cette limitation ne pose pas un réel problème étant donné que les signaux utilisés en sismique sont toujours à moyenne nulle (il est également possible de seuiller le spectre du filtre, pour qu'il n'y ait aucune valeur nulle).

On essaie en fait plusieurs formes de filtres rho où la puissance de la fréquence varie:

$$\hat{\rho}_\alpha(\omega) = |\omega|^\alpha \tag{E3.46}$$

On effectue des LRD avec un gradient préconditionné avec plusieurs valeurs de α (fig. 3.11a). La meilleure convergence est obtenue pour des valeurs de α entre 0.5 et 1.0, la valeur théorique étant 1.0. Dans la pratique on pourra donc prendre cette valeur, pour laquelle la convergence est environ 3 fois plus rapide que sans filtre rho ($\alpha=0$.)

Le filtre rho théorique pour la PRD est plus complexe que pour la LRD, puisqu'il agit non seulement par rapport au temps, mais également par rapport aux courbures (équ. (E1.54)). On peut néanmoins retenir une forme simplifiée en ne gardant que la composante temps. La meilleure convergence (fig. 3.11b) est obtenue pour des puissances entre 0.5 et 1.0, pour une valeur théorique de 0.5 que l'on pourra donc retenir en pratique. A la différence de la LRD, la convergence à la première itération est peu affectée par le préconditionnement. A partir de la deuxième itération, la convergence est environ 2 fois plus rapide avec le filtre rho.

Enfin il faut traiter le cas de la HRD: il n'est pas possible de calculer explicitement un filtre rho, même dans les conditions idéales (ouverture infinie, variables continues). Néanmoins, rien n'interdit d'essayer les mêmes filtres rho que pour la LRD et la PRD, une hyperbole pouvant être assimilée à une parabole à son apex et à une droite à son asymptote. La famille d'hyperboles utilisée est la famille P-SCAN (E1.7). La convergence (fig. 3.11c) est la meilleure pour une puissance de 0.5 (2 fois plus rapide que sans filtre rho). De même que pour la PRD, la première itération semble peu affectée par l'application du filtre rho.

En norme L2, le filtre rho de préconditionnement du gradient est:

$$\begin{aligned}
 &|\omega| \text{ pour la LRD} \\
 &|\omega|^{0.5} \text{ pour la PRD} \\
 &|\omega|^{0.5} \text{ pour la HRD}
 \end{aligned}$$

Discussion

On aurait pu craindre, sur des données réelles contenant du bruit, que la partie HF du bruit soit excessivement relevée par le filtre rho sur le gradient, affectant ainsi la convergence (Lumley, 1994). Or il semble que cela ne pose en fait pas de problème. Par prudence, il convient néanmoins de s'assurer que les fréquences au-delà du spectre du signal ont bien été filtrées sur les données.

L'utilisation de ces filtres rho pour des résolutions autres qu'en norme L2 (norme L1 ou autres) n'est pas souhaitable. En effet, la justification de leur utilisation est que l'opérateur adjoint suivi du filtre rho réalise une bonne approximation de l'inverse en norme L2. Or dans une résolution autre qu'en norme

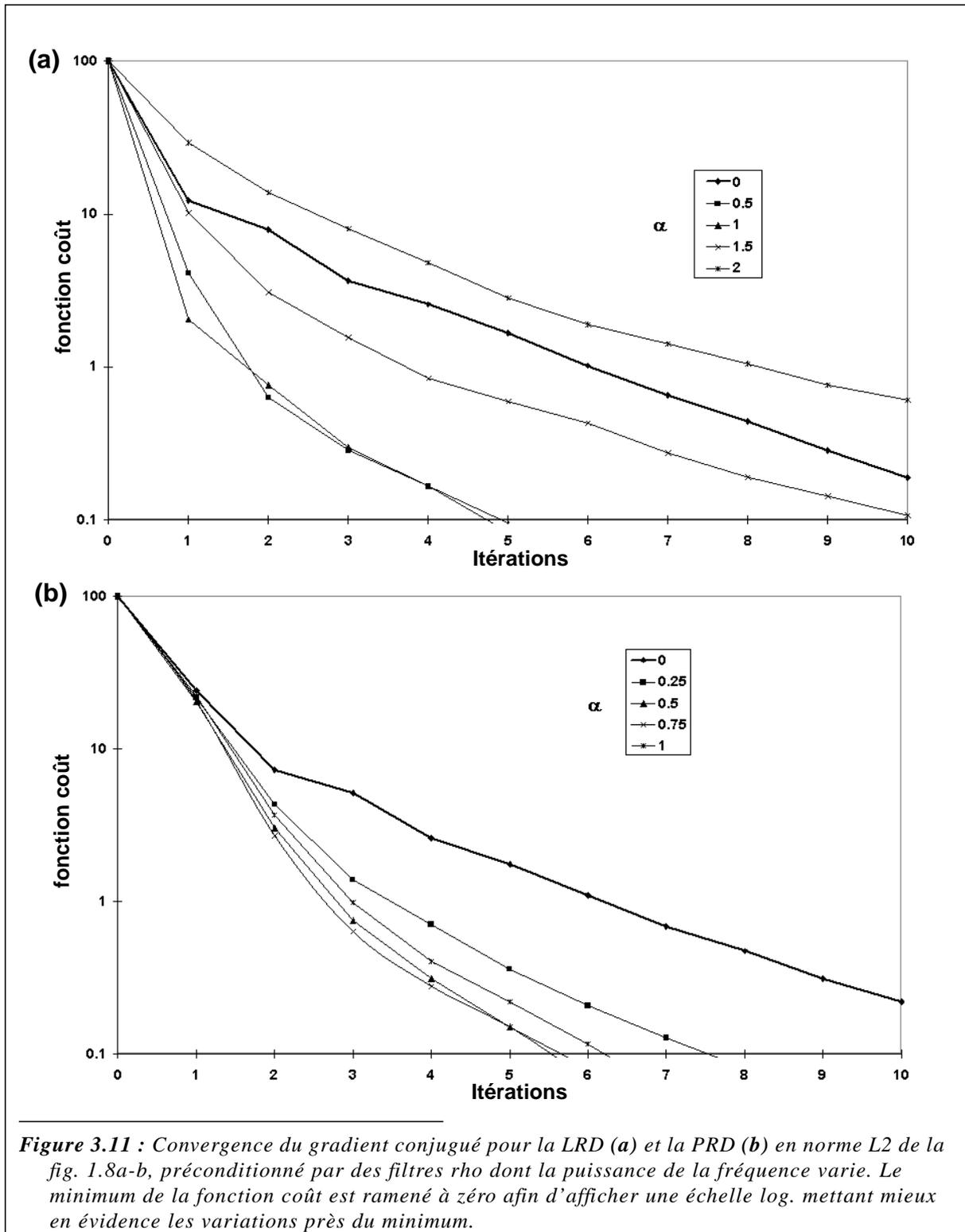
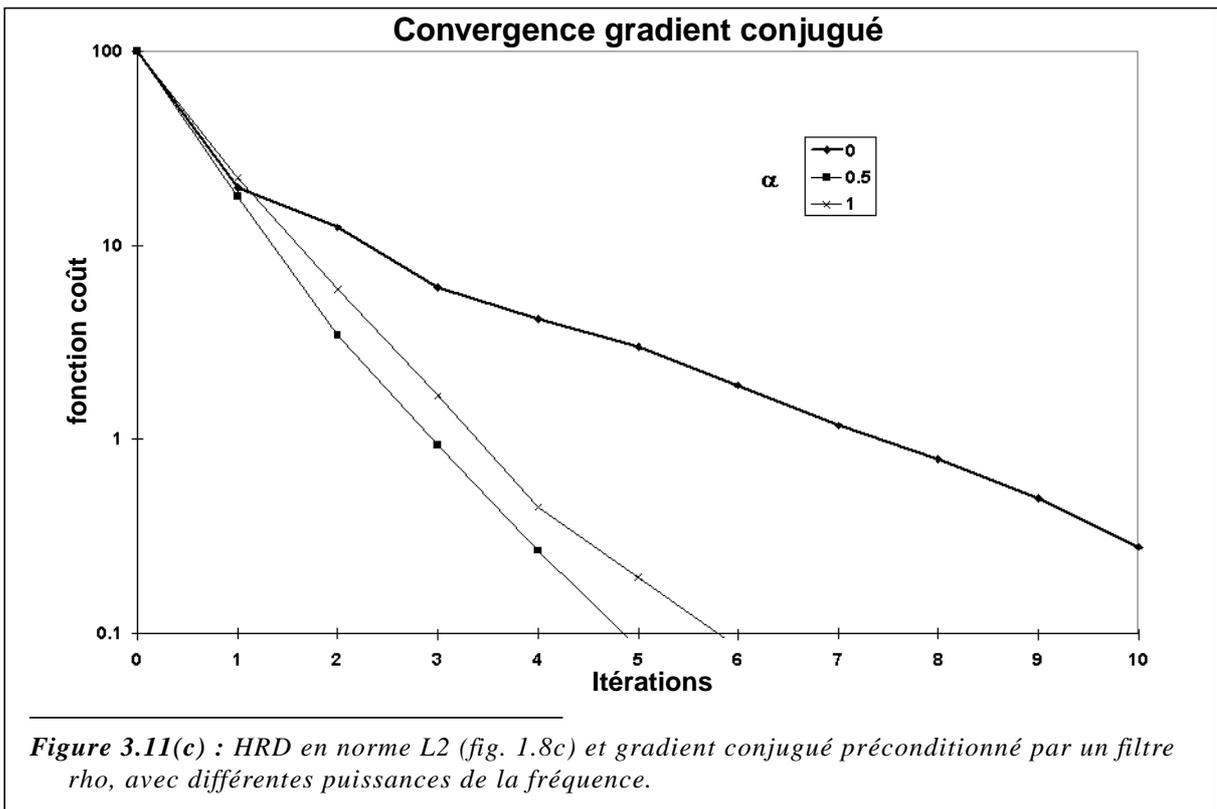


Figure 3.11 : Convergence du gradient conjugué pour la LRD (a) et la PRD (b) en norme L2 de la fig. 1.8a-b, preconditionné par des filtres rho dont la puissance de la fréquence varie. Le minimum de la fonction coût est ramené à zéro afin d'afficher une échelle log. mettant mieux en évidence les variations près du minimum.

L2, il n'y a pas à priori pas d'intérêt à vouloir approximer la solution en norme L2 (la convergence peut même en être ralentie).

Implémentation - performance

La longueur-temps des filtres rho étant assez importante, leur application se fait dans le domaine de Fourier, impliquant donc pour chaque trace du modèle une FFT et une FFT inverse. Sur un ordinateur scalaire classique le surcoût CPU du preconditionnement est d'environ 20%, à mettre en balance avec un nombre d'itérations 2 à 3 fois plus faible. Sur un ordinateur vectoriel (CRAY C92), par contre, le



surcoût atteint 40% (les FFT se vectorisant moins bien que les opérations de base du gradient conjugué (multiplications matrice*vecteur)), diminuant ainsi l'intérêt du préconditionnement, bien qu'il garde l'avantage.

3.8.2 - filtres rho plus sophistiqués

Les filtres rho utilisés dans le paragraphe précédent sont basés sur des conditions 'idéales': variables continues, domaines infinis. Il est possible d'affiner la détermination des filtres en tenant compte de la nature discrète et finie des transformées.

Le rôle du filtre rho est de compenser les atténuations fréquentielles dues à l'opérateur $\mathbf{L}^*\mathbf{L}$: les différentes composantes fréquentielles d'un même évènement sont en effet amplifiées différemment. Il s'agit à la base d'un problème de valeurs propres de $\mathbf{L}^*\mathbf{L}$. La dynamique de ces valeurs propres (le nombre de conditionnement du système) détermine la vitesse de convergence du gradient conjugué. Le filtre rho est un préconditionneur chargé de ramener le nombre de conditionnement le plus proche possible de 1. Nous cherchons donc à déterminer des vecteurs propres approchés et monofréquentiels, sans faire une recherche exacte ou exhaustive. On s'intéresse en fait à la variation des valeurs propres avec la fréquence, sans chercher forcément toutes les valeurs propres.

3.8.2.1 - LRD

Pour commencer, supposons que les variables sont continues, appartenant à des domaines bornés d'extension X et P centrés sur 0 (le centrage n'a pas d'importance, mais permet de simplifier les calculs).

$\underline{\mathcal{L}}^* \underline{\mathcal{L}}$ correspond, dans le domaine de Fourier, à une convolution suivant p à fréquence ω constante, par un signal connu $\sigma(\omega, p)$ (éq. (E1.52)), suivi par l'application d'une fonction 'porte' entre $-P/2$ et $P/2$. Exprimé dans les domaines (ω, p) et (ω, k_p) , cela donne:

$$\underline{\hat{\mathcal{L}}}^* \underline{\hat{\mathcal{L}}} \hat{\mathbf{u}}(\omega, p) = [\hat{\mathbf{u}}(\omega, p)^*_p \sigma(\omega, p)] \underset{-\frac{P}{2}}{\overset{P}{\Pi}}(p) \quad (\text{E3.47})$$

$$\underline{\hat{\mathcal{L}}}^* \underline{\hat{\mathcal{L}}} \hat{\mathbf{u}}(\omega, k_p) = [\hat{\mathbf{u}}(\omega, k_p) \sigma(\omega, k_p)]^*_{k_p} (P \text{sinc}(Pk_p/2))$$

L'équation (E1.53) donne pour l'opérateur de convolution et son spectre:

$$\sigma(\omega, p) = \int_{-X/2}^{X/2} e^{i\omega p x} dx = X \text{sinc}(\omega p X/2) \quad (\text{E3.48})$$

$$\sigma(\omega, k_p) = \frac{2\pi}{|\omega|} \left(\underset{-|\omega| \frac{X}{2}}{\overset{|\omega| \frac{X}{2}}{\Pi}}(k_p) \right)$$

Asymptotiquement, quand $\omega \rightarrow \infty$, on peut négliger l'effet de la porte $[-P/2$ et $P/2]$ dans (E3.47), et dans ce cas les valeurs propres sont les valeurs du spectre de l'opérateur de convolution: pour chaque fréquence ω , il y a deux valeurs propres: 0 et $\frac{2\pi}{|\omega|}$.

Lorsque $\omega \rightarrow 0$, la largeur du spectre de l'opérateur devient faible, et l'effet de la porte $[-P/2, P/2]$ ne peut être négligée. L'opérateur pour la fréquence nulle est constant:

$$\sigma(\omega=0, p) = X \quad (\text{E3.49})$$

Les deux seules valeurs propres sont donc PX et 0:

$$\begin{aligned} \underline{\hat{\mathcal{L}}}^* \underline{\hat{\mathcal{L}}} \hat{\mathbf{u}}(\omega=0, p) &= [\hat{\mathbf{u}}(0, p)^*_p \sigma(0, p)] \underset{-\frac{P}{2}}{\overset{P}{\Pi}}(p) \\ &= \left[X \int_{-P/2}^{P/2} \hat{\mathbf{u}}(0, p) dp \right] \underset{-\frac{P}{2}}{\overset{P}{\Pi}}(p) \end{aligned}$$

On décompose \mathbf{u} sur sa partie constante et son complémentaire:

$$\hat{\mathbf{u}}(\omega=0, p) = \alpha \underset{-\frac{P}{2}}{\overset{P}{\Pi}}(p) + \mathbf{V}(p) \quad \left(\mathbf{V}, \underset{-\frac{P}{2}}{\overset{P}{\Pi}}(p) \right) = \int_{-P/2}^{P/2} \mathbf{V}(p) dp = 0$$

D'où:

$$\underline{\hat{\mathcal{L}}}^* \underline{\hat{\mathcal{L}}} \hat{\mathbf{u}}(\omega=0, p) = (\alpha XP) \underset{-\frac{P}{2}}{\overset{P}{\Pi}}(p)$$

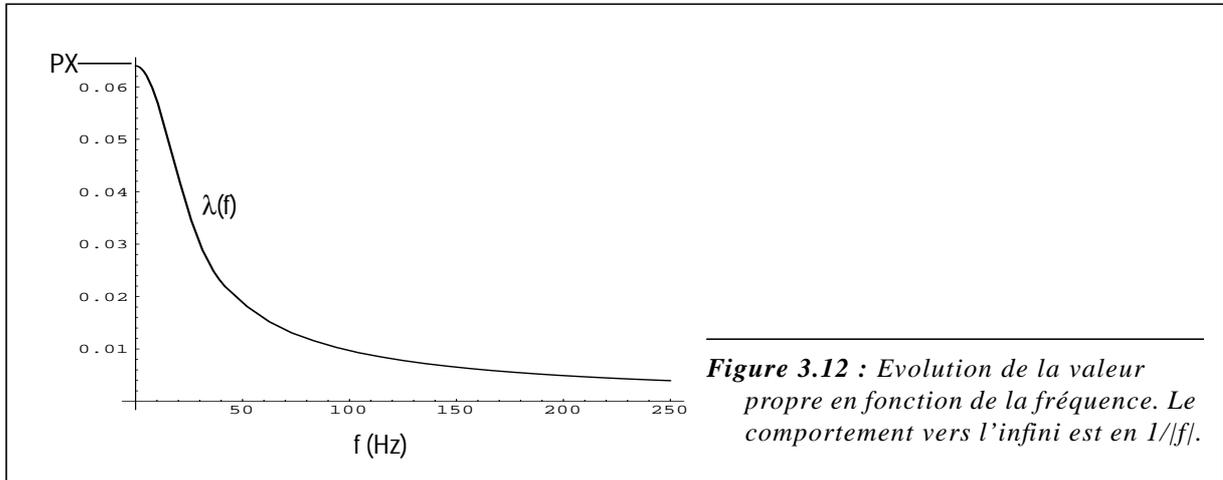
La porte $[-P/2, P/2]$ est donc le seul vecteur propre dont la valeur propre est non nulle, égale à PX .

En résumé, si P était infini les valeurs propres non nulles du système seraient $\lambda_{X, P=\infty}(\omega) = \frac{2\pi}{|\omega|}$ (une seule valeur propre par fréquence). Si P est fini, son effet est négligeable pour les hautes fréquences, et le comportement se rapproche de $\lambda_{X, P=\infty}(\omega)$; pour les basses fréquences, les valeurs propres tendent vers $\lambda_{X, P}(0) = PX$

3 -Le Problème inverse

La figure 3.12 montre les variations de cette valeur propre en fonction de la fréquence: son maximum vaut PX pour $f=0$, et son comportement à l'infini est en $1/|f|$. Le conditionnement du système, pour un contenu fréquentiel des données allant de 0 à f_{\max} , est donc approximativement égal à:

$$\frac{\lambda_{\max}}{\lambda_{\min}} \approx PXf_{\max} \quad (\text{E3.50})$$

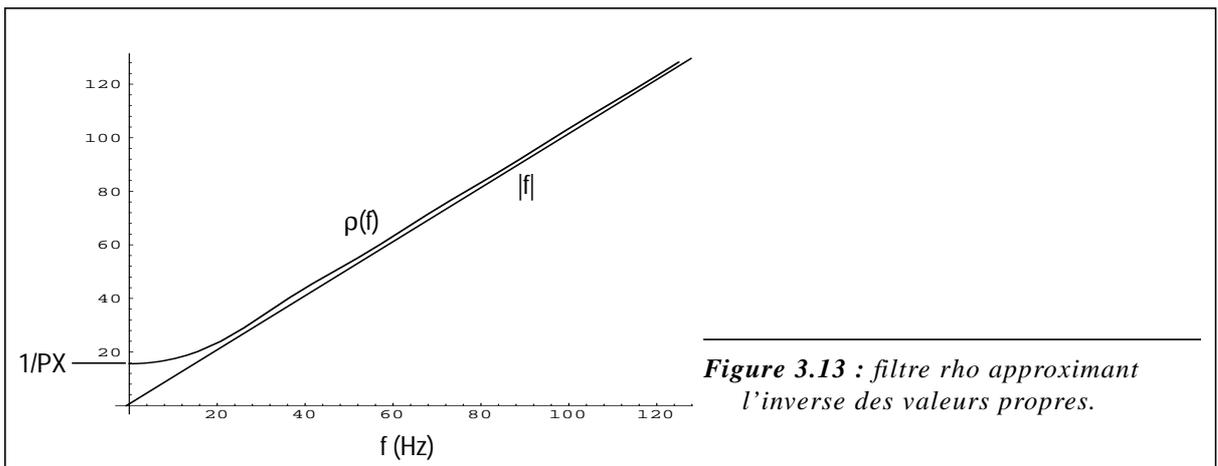


La convergence du gradient conjugué est d'autant plus lente que les gammes en x et p sont importantes et que la bande passante du signal est élevée.

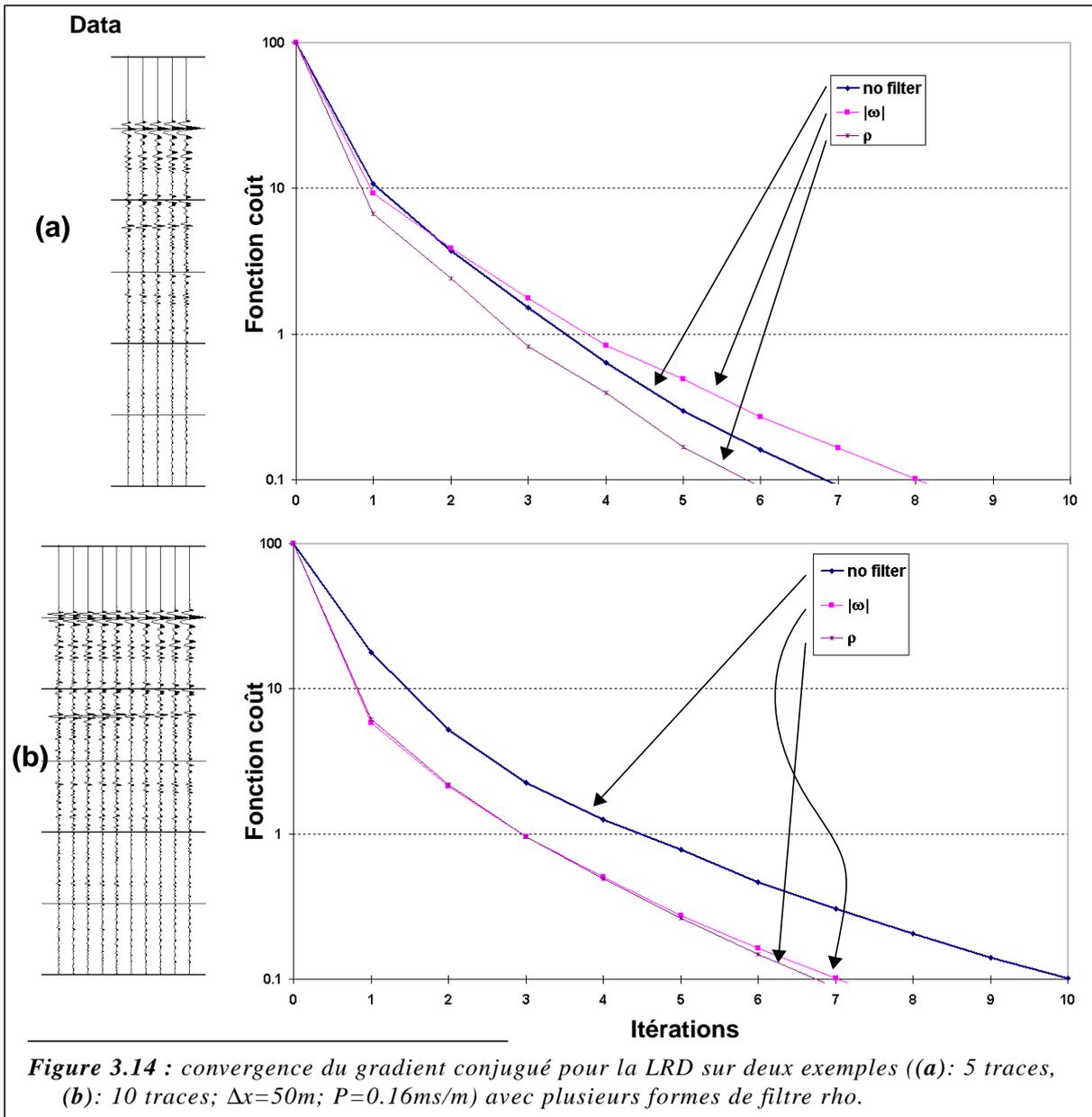
Un bon opérateur de préconditionnement doit appliquer à chaque fréquence une amplification inverse à la valeur propre pour cette fréquence. On approxime cette fonction par une hyperbole (fig 3.13):

$$\rho(\omega) \approx \frac{1}{\lambda(\omega)}$$

$$\rho(\omega) = \sqrt{\left(\frac{\omega}{2\pi}\right)^2 + \left(\frac{1}{PX}\right)^2} \quad (\text{E3.51})$$



On compare ce filtre au filtre classique en $|\omega|$ sur deux exemples: l'un de 5 traces et l'autre de 10 traces (mêmes intertraces $\Delta x=50\text{m}$) (fig. 3.14). Dans les deux cas: $P=0.160\text{ms/m}$. Avec 10 traces les deux filtres donnent des résultats équivalents, $1/PX$ étant négligeable par rapport aux fréquences considérées. Avec 5 traces, par contre, la convergence est nettement améliorée par rapport au filtre classique, qui dégrade même les résultats par rapport à la convergence sans filtre rho.



Après correction par le filtre rho approprié, les vitesses de convergence sont sensiblement les mêmes dans les deux exemples (dans les deux cas on est censé avoir égalisé les valeurs propres, donc ramené le nombre de conditionnement, qui détermine la vitesse de convergence, aux alentours de 1).

Dans les cas où un filtre rho est vraiment nécessaire pour accélérer la convergence (produit $f_{\max}PX$ important), le filtre classique donne donc d'aussi bons résultats que le filtre amélioré, et là où ce dernier donne de meilleurs résultats, la convergence était déjà relativement bonne sans filtre. Il semblerait donc inutile de chercher à utiliser le filtre amélioré. C'est relativement vrai pour la LRD 2D (τ, p) présentée ici, mais ce filtre amélioré sera par contre très utile pour le problème de la décomposition 3D (τ, p_x, p_y), dans lequel les problèmes de convergence sont beaucoup plus sensibles (voir le chapitre 6).

Aliasing spatial

Les calculs ci-dessus sont effectués avec des variables continues. La discrétisation ne pose pas de réel problème, sauf en cas d'aliasing spatial: des lobes secondaires apparaissent sur les opérateurs de convolution (fig. 2.10), modifiant les vecteurs et valeurs propres.

Le spectre de l'opérateur pour la variable discrétisée ($x_j, j=1..N_x$) est une succession de diracs positionnés d'après les x_j (équ. (E2.9)):

$$\sigma(\omega, k_p) = \sum_{x_j} 2\pi\delta(\omega x_j - k_p)$$

Pour $\omega = 0$, tous les diracs sont à la même position $k_p=0$, et l'opérateur est le même que dans le cas continu, à une constante multiplicative près:

$$\sigma(\omega=0, k_p) = 2\pi N_x \delta(k_p)$$

$$\sigma(\omega=0, p) = N_x$$

Les seules valeurs propres sont donc 0 et PN_x .

Quand $\omega \rightarrow \infty$, les diracs sont de plus en plus écartés les uns des autres, et à partir d'un certain point n'interagissent plus entre eux (après application de la porte $[-P/2, P/2]$, chaque dirac dans le domaine k_p est 'remplacé' par un sinus cardinal de demi-largeur $2\pi/P$ ne dépendant pas de la fréquence (E3.47); si les diracs sont suffisamment écartés, le sinus cardinal engendré par un dirac est négligeable au niveau des autres diracs). Il faut noter que la situation d'aliasing spatial $\omega\Delta x > 2\pi/P$ (équ. (E2.5)) correspond à une distance entre les diracs supérieure à la demi-largeur du sinus cardinal.

Dans ces conditions, tout signal $e^{i\omega x_j p} \frac{P}{2} \Pi_{\frac{P}{2}}(p)$, dont le spectre est $\text{Psinc}\left(\frac{P(k_p - \omega x_j)}{2}\right)$ (sinus cardinal centré sur un des diracs du spectre de l'opérateur et de demi-largeur $2\pi/P$) est un vecteur propre, dont la valeur propre est P:

$$\begin{aligned} \hat{\mathbf{L}}^* \hat{\mathbf{L}} \left(\text{Psinc}\left(\frac{P(k_p - \omega x_j)}{2}\right) \right) (\omega, k_p) &= \left[\text{Psinc}\left(\frac{P(k_p - \omega x_j)}{2}\right) \sigma(\omega, k_p) \right] *_{k_p} \left(\text{Psinc}\left(\frac{Pk_p}{2}\right) \right) \\ &\approx [P\delta(k_p - \omega x_j)] *_{k_p} \left(\text{Psinc}\left(\frac{Pk_p}{2}\right) \right) \\ &= P^2 \text{sinc}\left(\frac{P(k_p - \omega x_j)}{2}\right) \end{aligned}$$

Les valeurs propres significatives ne tendent donc pas vers 0 comme quand x est continue, mais sont minorées par P. Lorsque la fréquence augmente, la valeur propre maximum pour cette fréquence décroît de PN_x à P, puis se stabilise autour de cette valeur. Cette borne inférieure sur les valeurs propres implique une borne supérieure sur le nombre de conditionnement du système, égale au nombre de traces dans les données:

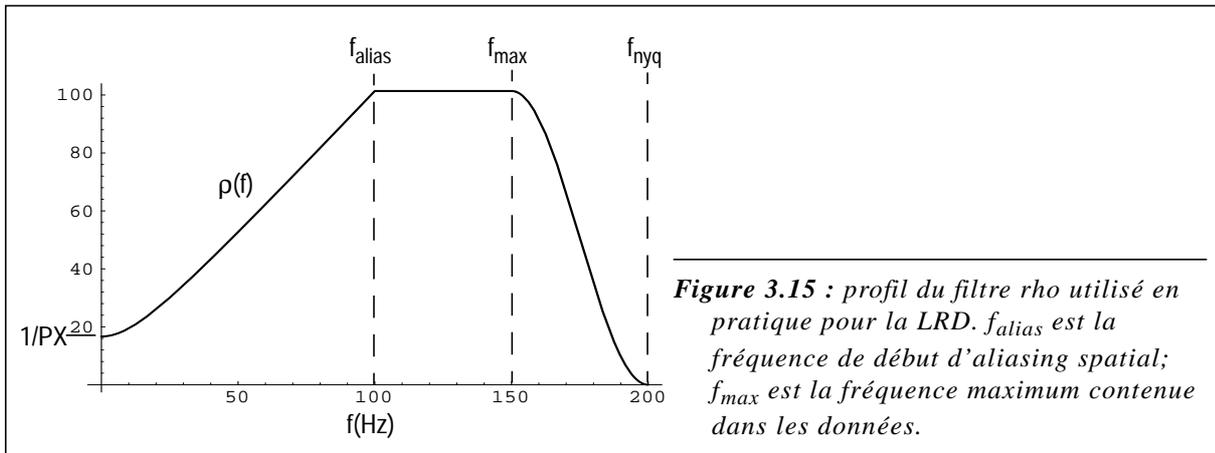
$$c = \frac{\lambda_{\max} = PN_x}{\lambda_{\min} \geq P} \leq N_x \quad (\text{E3.52})$$

Il est clair que lorsque l'on applique les filtres rho classiques, non bornés vers les hautes fréquences, on amplifie exagérément les fréquences spatialement aliasées. Pour éviter cela, on plafonne la courbe du filtre rho par sa valeur à la fréquence de début d'aliasing. Par ailleurs, on introduit une rampe (en \cos^2)

3.8 -Préconditionnement par un filtre rho

sur les HF à partir de la fréquence maximum des données jusqu'à la fréquence de Nyquist, afin de pas trop amplifier le bruit numérique (qui est HF). Le filtre rho utilisé en pratique est donc (tracé fig. 3.15):

$$\rho(\omega) = \begin{cases} |\omega| < \omega_{\text{alias}} & \sqrt{\left(\frac{\omega}{2\pi}\right)^2 + \left(\frac{1}{PX}\right)^2} \\ \omega_{\text{alias}} < |\omega| < \omega_{\text{max}} & \sqrt{\left(\frac{\omega_{\text{alias}}}{2\pi}\right)^2 + \left(\frac{1}{PX}\right)^2} \\ \omega_{\text{max}} < |\omega| < \omega_{\text{nyq}} & \sqrt{\left(\frac{\omega_{\text{alias}}}{2\pi}\right)^2 + \left(\frac{1}{PX}\right)^2} \cos^2\left[\frac{\pi}{2} \frac{|\omega| - \omega_{\text{max}}}{\omega_{\text{nyq}} - \omega_{\text{max}}}\right] \end{cases} \quad (\text{E3.53})$$



3.8.2.2 - PRD et HRD

Le cas parabolique est très semblable au cas linéaire. Par contre pour le cas hyperbolique, il n'est pas possible de mener la même étude dans le domaine de Fourier.

En parabolique, prenons des domaines d'extension X (tel que $0 \leq x_{\text{min}} \leq x_{\text{max}} = x_{\text{min}} + X$, les offsets négatifs et positif étant indifférenciés en parabolique) et Q. L'opérateur de convolution et son spectre sont (équ. (E1.53)):

$$\sigma(\omega, q) = \int_{x_{\text{min}}}^{x_{\text{max}}} e^{i\omega qx^2} dx \quad (\text{E3.54})$$

$$\sigma(\omega, k_q) = \frac{\pi}{\sqrt{|\omega|k_q}} \left(\Pi_{|\omega|x_{\text{min}}^2} \Pi_{|\omega|x_{\text{max}}^2} (k_q) \right)$$

Le spectre, donc les valeurs propres, pour une fréquence ω donnée varient avec k_q . Pour chaque fréquence il faut en fait comparer les informations qui proviennent de la même zone: la modélisation s'exprime ainsi (équ. (E1.44)):

$$\begin{aligned}
 d(\omega, x) &= \int_q u(\omega, q) e^{-i\omega q x^2} dq \\
 &= \int_q \frac{1}{2\pi} \int_{k_q} u(\omega, k_q) e^{iqk_q} dk_q e^{-i\omega q x^2} dq \\
 &= \frac{1}{2\pi} \int_{k_q} u(\omega, k_q) \int_q e^{iq(k_q - \omega x^2)} dq dk_q \\
 &= \frac{Q}{2\pi} \int_{k_q} u(\omega, k_q) \text{sinc}(Q(k_q - \omega x^2)/2) dk_q
 \end{aligned}$$

Cela signifie que l'information nécessaire à la reconstruction des données au point (ω, x) se situe autour du point $(\omega, k_q = \omega x^2)$. Il faut donc observer la variation avec la fréquence du spectre de l'opérateur en ce point:

$$\sigma(\omega, k_q = \omega x^2) = \frac{\pi}{\sqrt{|\omega \omega x^2|}} = \frac{\pi}{|\omega x|} \quad (\text{E3.55})$$

De la même façon que pour la LRD, ces valeurs sont des valeurs propres dans le cas où Q est infini, ou pour $|\omega|$ très grand. On montre également de la même manière que pour la LRD que pour la fréquence nulle, la seule valeur propre non nulle est QX .

Le comportement asymptotique est donc en $1/|\omega|$, comme pour la LRD. Le problème vient ici de la dépendance en x des valeurs propres qui rend difficile la définition d'un filtre ρ tenant compte de la limite QX en $\omega = 0$ (indépendante de x). On peut par exemple choisir de définir le filtre par rapport à l'offset moyen:

$$\rho(\omega) = \sqrt{\left(\frac{\omega(x_{\min} + x_{\max})}{2\pi}\right)^2 + \left(\frac{1}{QX}\right)^2} \quad (\text{E3.56})$$

Cela revient en fait à considérer la PRD comme une LRD dans laquelle on utilise la variable x^2 à la place de x , et à appliquer le filtre ρ correspondant défini en (E3.51). On trouverait en effet:

$$\begin{aligned}
 \rho(\omega) &= \sqrt{\left(\frac{\omega}{2\pi}\right)^2 + \left(\frac{1}{Q(x_{\max}^2 - x_{\min}^2)}\right)^2} \\
 &= \sqrt{\left(\frac{\omega}{2\pi}\right)^2 + \left(\frac{1}{Q(x_{\max} - x_{\min})(x_{\max} + x_{\min})}\right)^2} \\
 &= \frac{1}{(x_{\max} + x_{\min})} \sqrt{\left(\frac{\omega(x_{\max} + x_{\min})}{2\pi}\right)^2 + \left(\frac{1}{QX}\right)^2}
 \end{aligned}$$

Les mêmes rampes que pour la LRD peuvent être introduits pour tenir compte de l'aliasing spatial et de la fréquence maximum des données.

Discussion

Comme pour la LRD, les résultats obtenus avec ce filtre s'écartent peu de ceux obtenus avec le simple filtre en $|\omega|$, sauf pour des données d'extension X très limités. On note au passage que le filtre théorique pour la PRD est $|\omega|$, et non pas $\sqrt{|\omega|}$, comme écrit aux parag. 1.9.5.1 et 3.8.1 (pour prendre partiellement en compte la variation en $1/\sqrt{|k_q|}$ du spectre de l'opérateur de convolution). Néanmoins, le filtre en $\sqrt{|\omega|}$ permet d'obtenir des résultats proches de l'optimum dans tous les cas de figures. C'est donc celui-ci qui est utilisé en pratique.

Concernant la HRD, le même type de calcul dans le domaine fréquentiel étant impossible, on ne peut que s'en tenir au filtre en $\sqrt{|\omega|}$, expérimentalement le plus efficace.

3.9 - Conclusions

L'application des théories du problème inverse à la décomposition de Radon permet de bien poser le problème et de comprendre la signification des paramètres utilisés.

Le problème est ici posé sous une forme proche des moindres carrés généralisés, avec des poids directement liés aux matrices de covariances, conduisant à la minimisation d'une forme quadratique (si les poids sont constants) ou d'une fonction approximable par des formes quadratiques (si les poids sont eux-mêmes fonctions du modèle).

Le gradient conjugué est une technique de minimisation bien adaptée pour ce problème. Si les poids ne sont pas constants, nous avons vu 2 façons d'utiliser le gradient conjugué dans le cadre des algorithmes IRLS.

Dans l'étude détaillée des décompositions de Radon en moindres carrés, il apparaît que le préconditionnement par un filtre rho permet d'accélérer la convergence du gradient conjugué. Le filtre rho pour la LRD proposé ici est construit en fonction des ouvertures et des pas d'échantillonnages des différentes variables, et est efficace dans tous les cas de figure (ce n'est pas le cas des filtres rho classiques, déduits de la transformée continue et d'ouverture infinie). Dans une certaine mesure, on peut construire le même type de filtre pour la PRD.

Nous avons par ailleurs mis en évidence que, contrairement à une idée reçue, les décompositions de Radon en moindres carrés ne permettent pas de filtrer le bruit aléatoire, même si la valeur du facteur d'amortissement doit être déterminée d'après le niveau estimé de celui-ci.

4 - Solutions Parcimonieuses

4.1 - Introduction

La Décomposition de Radon, qu'elle soit Linéaire, Parabolique, ou Hyperbolique, est la plupart du temps effectuée en norme L2 (équ. (E3.36) et (E3.37)), pour des raisons qui tiennent aussi bien à la facilité d'implémentation (algorithmes maintenant bien connus), à l'efficacité (temps de calcul réduit, souvent par une implémentation fréquentielle), et à une certaine facilité (peu de réglages des paramètres, qui sont en outre peu sensibles).

Néanmoins la norme L2 a certaines limitations, plus ou moins bien mises en évidence dans la littérature: étalement de la représentation dans le domaine transformé (donc séparabilité moyenne des évènements), sensibilité aux bruits forts et isolés (spikes), impossibilité de traiter l'aliasing spatial notamment.

Aussi, dans certains cas il peut être souhaitable d'appliquer d'autres critères que la norme L2, aussi bien sur les données que sur le modèle.

Thorson and Claerbout (1985) ont présenté plusieurs méthodes pour résoudre la Décomposition de Radon, (dont ils ont d'ailleurs introduit le concept de calcul par inversion dans le même article): la simple application de l'opérateur adjoint comme approximation de l'inverse, la résolution classique en norme L2, et une inversion dite 'stochastique', en introduisant un à priori sur les variances du modèle, elles-mêmes déduites du modèle estimé en cours de calcul, définissant ainsi un processus dit 'bootstrappé'.

Nichols (1994) applique un critère de minimisation de la norme Lp (p compris entre 1 et 2) du modèle au lieu de la norme L2, afin que la solution obtenue se conforme à des distributions autres que gaussiennes. Le même principe est d'ailleurs applicable à la minimisation des résidus.

Plus récemment Sacchi and Ulrych (1995) ont présenté une solution très proche de celle de Thorson and Claerbout (1985), en autorisant plus de souplesse et liberté dans le choix des paramètres.

Ces méthodes ont toutes en commun de conduire à des solutions dites '*parcimonieuses*', qui favorisent la concentration de l'énergie en certains points du modèle, et laissant des valeurs faibles en dehors de ces points. Cette propriété est souvent désirée pour la Décomposition de Radon, puisque l'on voudrait qu'un évènement 'pur' soit représenté en seul point du modèle, ce que ne permet pas de faire la norme L2.

La parcimonie (sparseness en anglais) est une notion qui ne possède pas de définition mathématique bien précise. Elle qualifie généralement la rareté des valeurs non nulles dans un ensemble donné, ou encore la non-uniformité de la répartition des valeurs dans une distribution. Elle est souvent reliée à la notion d'entropie (Thorson and Claerbout, 1985; Carron, 1989). L'entropie associée à une variable aléatoire discrète a été définie par Shannon (1948):

$$H = - \sum_{i=1}^N p_i \log p_i \quad (E4.1)$$

où N est le nombre d'évènements possibles et p_i la probabilité de réalisation de l'évènement i. On vérifie facilement que l'entropie est minimum ($H=0$) pour un seul évènement possible ($N=1$), et maximum ($H=\log N$) pour une distribution uniforme des probabilités. On est proche de la notion de parcimonie, opposée à l'entropie. Une autre mesure de la parcimonie est le varimax (Wiggins, 1985), ou plutôt les varimax, d'une variable aléatoire x sur n réalisations:

$$V_2(x) = \frac{\sum x_i^2}{(\sum |x_i|)^2} \quad V_4(x) = \frac{\sum x_i^4}{(\sum x_i^2)^2} \quad (E4.2)$$

Les varimax sont compris entre $1/n$ (n éléments tous égaux) et 1.0 (un seul élément non nul).

Les minimisations en norme L2 ne conduisent pas à des solutions parcimonieuses, car elles pénalisent trop les valeurs fortes, et privilégient plutôt des répartitions gaussiennes. Toute minimisation qui tend à favoriser, par rapport à la norme L2, les valeurs extrêmes (fortes ou faibles) pourra être qualifiée de parcimonieuse. C'est le cas pour la norme Lp (avec $p < 2$) et la solution de Thorson.

Ces problèmes, tels qu'ils sont posés par ces auteurs, sont non linéaires. Ils sont généralement résolus par des algorithmes itératifs de type IRLS (Iteratively Reweighted Least Squares), dans lesquels la fonction à minimiser est approximée localement par une forme quadratique ré-estimée au fur et à mesure des itérations. On minimise donc des formes quadratiques successives, généralement par des méthodes de gradient (simple ou conjugué...). Ces résolutions réclament un nombre d'itérations supérieur à la résolution en norme L2 (qui constitue un unique problème quadratique), surcoût pouvant être prohibitif lorsqu'il s'agit de traiter des gros volumes de données.

La justification de la minimisation de telle ou telle quantité (norme L2, norme L1, entropie...) n'est pas claire. Ce choix devrait normalement être guidé par le type de distribution des valeurs sur la solution cherchée: la minimisation de la norme L2 suppose une distribution gaussienne des valeurs, la norme L1 une distribution exponentielle... Or la minimisation elle-même influe fortement sur la forme de la solution trouvée (on ne trouve que ce que l'on cherche). Il faudrait donc avoir par d'autres moyens un à priori sur la forme générale de la solution, comme c'est le pour la déconvolution: le modèle est une suite de coefficients de réflexion, dont on peut connaître la distribution en analysant les logs soniques aux puits (Carron, 1989). Malheureusement, dans le cas des Décompositions de Radon, on ne dispose pas de ce type de renseignement.

Enfin, les solutions parcimonieuses se révèlent beaucoup plus sensibles que la norme L2 aux réglages de certains paramètres, notamment l'estimation du niveau du bruit sur les données: si celui-ci est sous-estimé la solution obtenue est trop bruitée, et si il est surestimé des petits événements risquent d'être assimilés à du bruit et de ne pas apparaître sur la solution. Ce problème est peu évoqué dans la littérature, mais il semble néanmoins fondamental dans les applications industrielles traitant des gros volumes de données, pour lesquelles on ne peut se permettre de se focaliser sur le réglage fin de paramètres aussi sensibles (et dépendants des données), et encore moins de perdre de l'information.

On étudie ici plus en détail les solutions obtenues par minimisation de la norme Lp ($1 < p \leq 2$) du modèle. Cette minimisation peut être reformulée en moindres carrés généralisés par l'utilisation d'un poids non plus constant mais dépendant directement du modèle lui-même. On met en évidence que ce poids doit être lisse à l'échelle de la longueur du signal sismique, et on compare deux implémentations possibles de l'algorithme IRLS. Nous étudions également la sensibilité de l'estimation du niveau de bruit (donc du réglage de l'importance du terme de régularisation).

La minimisation de la norme Lp des résidus ne conduit pas à une solution parcimonieuse mais permet de mieux traiter les bruits non gaussiens sur les données (des bruits isolés énergiques par exemple). La convergence du gradient conjugué est dégradée, et un préconditionnement adapté doit être appliqué pour l'améliorer.

En reprenant les principes de base de la solution de Thorson and Claerbout (1985) nous proposons une solution parcimonieuse qui repose sur l'estimation du signal et du bruit le long de chacune des courbes de la famille utilisée. Cette solution reste raisonnable en termes de temps de calcul et surtout évite d'avoir à déterminer à priori le niveau de bruit, paramètre sensible dans toutes les décompositions parcimonieuses.

4.2 - Application d'une Norme Lp sur le modèle.

4.2.1 - Présentation

L'application d'une norme Lp sur le modèle signifie que l'on régularise l'inversion en minimisant la norme Lp du modèle (avec $1 < p \leq 2$) au lieu de la norme L2 dans l'équation (E3.36). La norme Lp d'un modèle $\mathbf{u}(t_0, x_0)$ étant:

$$\|\mathbf{u}\|_{Lp} = \left(\iint_{t_0 x_0} |\mathbf{u}(t_0, x_0)|^p dt dx \right)^{\frac{1}{p}} \quad (\text{E4.3})$$

Cette expression n'est à strictement parler une norme que pour p supérieur ou égal à 1.

Dans l'optique bayésienne de l'inversion, minimiser la norme Lp revient à trouver le modèle le plus probable dans le cas où la pdf liée à ce modèle est une distribution gaussienne généralisée d'ordre p (Tarantola, 1987).

Pour un problème à une seule variable u, la gaussienne généralisée d'ordre p, centrée sur u0 et de dispersion σ_p est donnée par (voir les courbes correspondantes sur la figure 4.1):

$$g_p(u) = \frac{p^{1-\frac{1}{p}}}{2\sigma_p \Gamma(1/p)} e^{-\frac{1}{p} \left| \frac{u-u_0}{\sigma_p} \right|^p} \quad (\text{E4.4})$$

($\Gamma(\)$ désigne la fonction Gamma).

Comparée à la distribution gaussienne (p=2), la distribution double exponentielle (p=1) favorise les valeurs extrêmes (petites ou grandes) au détriment des valeurs moyennes. Cette propriété est intéressante dans le cas de données comportant des événements purs (conformes au modèle): on souhaiterait à la limite trouver un modèle contenant des diracs, que l'on peut approcher par un modèle contenant quelques échantillons significatifs parmi une majorité d'échantillons de valeurs négligeables. Or la norme L2 tend à lisser le modèle, en «gommant» les pics pour répartir l'énergie sur un plus grand nombre d'échantillons de valeurs plus faibles, afin de le conformer à la distribution gaussienne associée.

4.2.2 - Implémentation, algorithme IRLS

Par rapport à l'inversion en norme L2 (équ. (E3.36)), on cherche cette fois à minimiser la quantité:

$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|_{L2}^2 + \frac{\varepsilon}{2} \|\mathbf{u}\|_{Lp}^p \quad (\text{E4.5})$$

Ce problème ne peut plus s'exprimer sous une forme quadratique, le terme de régularisation $\frac{\varepsilon}{2} \|\mathbf{u}\|_{Lp}^p$ ne pouvant être développé en un produit scalaire. On cherche donc à le linéariser en approximant

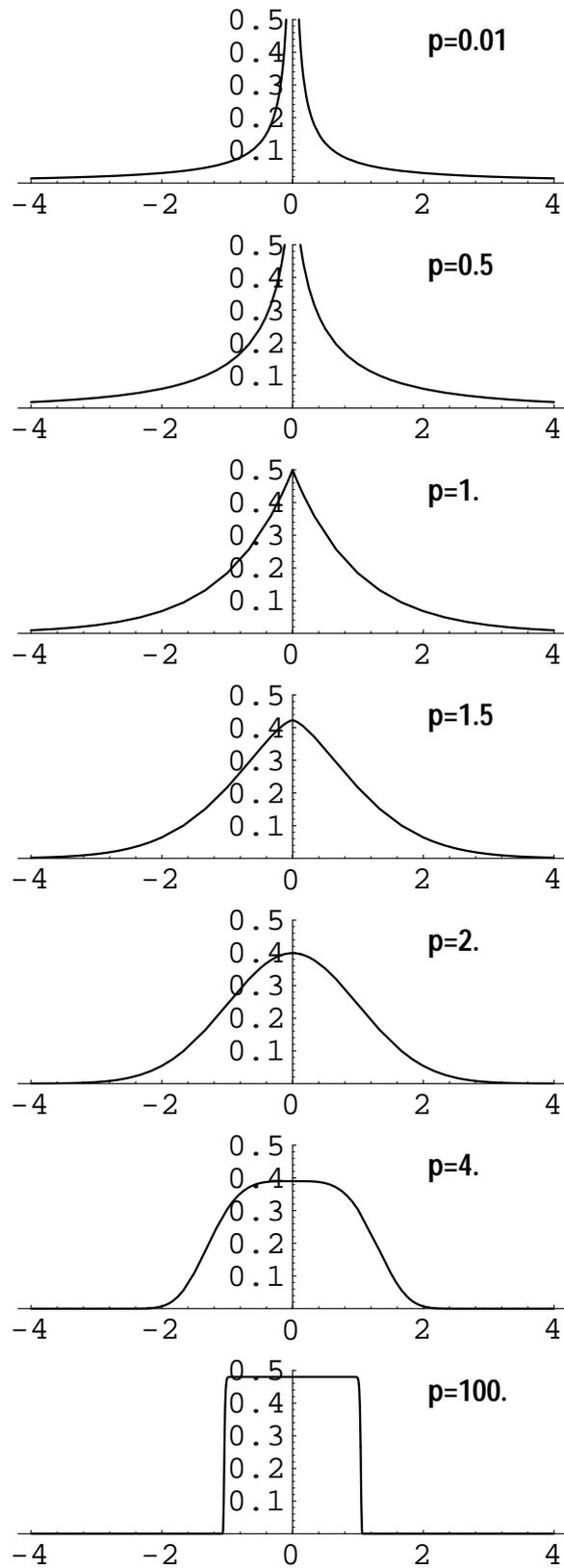


Figure 4.1 : Gaussiennes généralisées d'ordre p , centrée en 0 et de déviation normalisée 1. $p=2$. donne la gaussienne classique, $p=1$ la distribution double exponentielle, $p=\infty$ une «box-car», $p=0$. un dirac.

4.2 -Application d'une Norme Lp sur le modèle.

localement la norme Lp par un produit scalaire pondéré par un poids (paragr. 3.5.3.2 et annexe B), afin d'appliquer un algorithme IRLS:

$$f(\mathbf{u}) = |\mathbf{u}|^p$$

$$(\mathbf{P}_{Lp}(\mathbf{u}))_{ii} = \frac{f'(u_i)}{u_i} = \frac{p|u_i|^{p-1}}{|u_i|} = \frac{p}{|u_i|^{2-p}} \quad (\text{E4.6})$$

Le poids \mathbf{P}_{Lp} est diagonal et fonction de \mathbf{u} . Afin de vérifier la condition de continuité (E3.35)(i) pour la convergence, il est nécessaire de modifier légèrement la définition au voisinage de 0:

$$\boxed{(\mathbf{P}_{Lp}(\mathbf{u}))_{ii} = \frac{p}{(|u_i| + \eta)^{2-p}}} \quad (\text{E4.7})$$

Où η est une petite valeur (par rapport aux valeurs significatives du modèle) strictement positive, souvent définie par rapport à la valeur maximum du modèle:

$$\eta = \frac{(\max|\mathbf{u}|)}{\lambda} \quad (\text{E4.8})$$

Avec comme valeur typiquement utilisée:

$$\lambda = 100 \quad (\text{E4.9})$$

La dynamique du poids dépend directement de λ :

$$\frac{(\max(\mathbf{P}_{Lp}))}{(\min(\mathbf{P}_{Lp}))} = \frac{1/\eta^{2-p}}{1/((\max|\mathbf{u}|) + \eta)^{2-p}} = \left(\frac{((\max|\mathbf{u}|) + \eta)}{\eta}\right)^{2-p} = (\lambda + 1)^{2-p} \quad (\text{E4.10})$$

Les autres conditions de convergence dans (E3.35) ne sont vérifiées que pour $1 < p \leq 2$ (en fait, $p=1$ est possible une fois que l'on a introduit η).

On doit donc maintenant résoudre un problème exprimé sous une forme quadratique, nous ramenant ainsi au problème général exprimé par les équations (E3.17) et (E3.18):

$$\psi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|_{L2}^2 + \frac{\varepsilon}{2} (\mathbf{u}, \mathbf{P}_{Lp}(\mathbf{u})\mathbf{u}) \quad (\text{E4.11})$$

Les échantillons du modèle sont affectés d'une pondération d'autant plus faible que leurs valeurs sont élevées. Une régularisation en norme Lp ($p < 2$) pénalise donc relativement plus les échantillons déjà faibles que la norme L2, et laisse plus de possibilités à des pics isolés d'apparaître.

On utilise ici le même processus IRLS que Nichols (1994), à savoir un gradient conjugué à l'intérieur duquel on effectue une mise à jour des poids, ici à chaque itération. On préconditionne par ailleurs le gradient par \mathbf{P}_{Lp}^{-1} , ceci afin de favoriser les échantillons de faible poids (ce préconditionnement a une autre justification: la minimisation de (E4.11) conduit à une solution qui n'a pas d'énergie dans le noyau de $\mathbf{L}\mathbf{P}_{Lp}^{-1}$ (paragr. 3.4). Il est donc souhaitable que la direction de descente possède la même propriété, ce qui s'obtient en appliquant l'opérateur adjoint $(\mathbf{L}\mathbf{P}_{Lp}^{-1})^* = \mathbf{P}_{Lp}^{-1}\mathbf{L}^*$ aux résidus):

$$\boxed{\mathbf{g}_p(\mathbf{u}) = \mathbf{P}_{Lp}^{-1}(\mathbf{L}^*\mathbf{r} - \varepsilon\mathbf{P}_{Lp}\mathbf{u}) = \mathbf{P}_{Lp}^{-1}\mathbf{L}^*\mathbf{r} - \varepsilon\mathbf{u}} \quad (\text{E4.12})$$

4.2.3 - Exemple

On effectue la PRD en norme L_p d'une collection CMP (fig. 4.2a). η est fixé de manière à ce que la dynamique du poids soit égale à 100, et ϵ est calculé itérativement afin d'atteindre un à priori de 5% de résidus. La comparaison des PRD en norme L_2 , $L_{1.5}$, et L_1 (fig. 4.2 b, c et d) montre une focalisation croissante des événements lorsque p diminue. L'énergie des résidus est comprise entre 2% et 3% pour ces trois décompositions.

Le calcul des varimax confirme le caractère parcimonieux des normes L_p ($p < 2$):

Tableau 1 : Varimax des décompositions en norme L_p

Décomposition	$\sum u_i^2$	$\sum u_i $	$V_2(\mathbf{u})$
Norme L_2	0.113	18.4	$3.3 \cdot 10^{-4}$
Norme $L_{1.5}$	0.126	12.8	$7.7 \cdot 10^{-4}$
Norme L_1	0.180	10.1	$1.8 \cdot 10^{-3}$

4.2.4 - Lissage du poids

Le poids est calculé échantillon par échantillon de manière indépendante. Or des échantillons voisins sur l'axe du temps ne peuvent pas être considérés comme indépendants, et appliquer des poids trop différents dans un intervalle de temps inférieur à la longueur de l'ondelette sismique risque de conduire à une déformation de celle-ci.

Le poids sur le modèle traduit l'inverse de la matrice de covariance sur le modèle (équ. (E3.16)). Celle-ci est diagonale si les échantillons sont indépendants les uns des autres. Or Des échantillons proches sont corrélés, ce qui se traduit par des coefficients non diagonaux.

Soit par exemple le problème très simple:

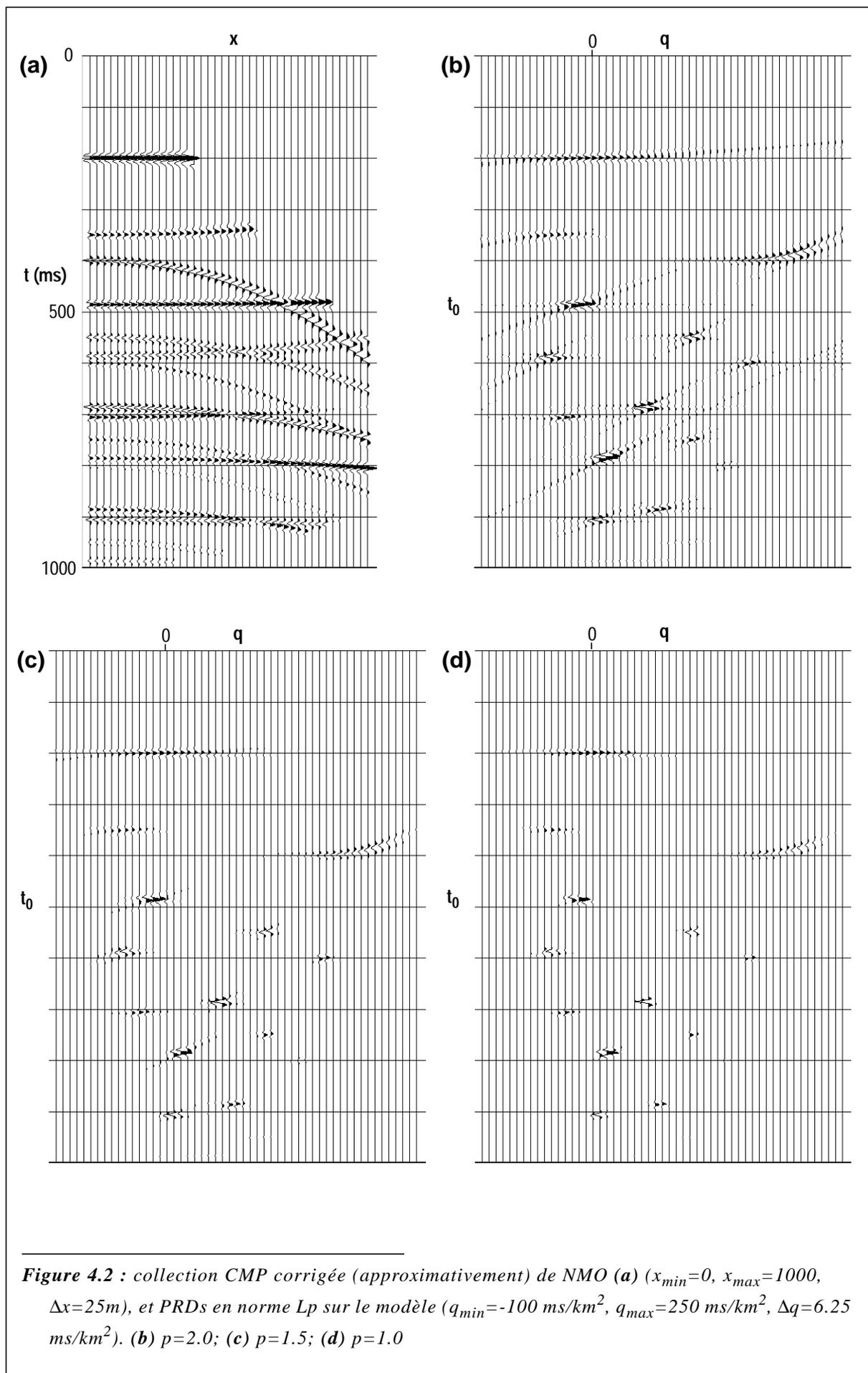
$$\mathbf{d} = \mathbf{L}\mathbf{u} = \begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 2u_1 + u_3 \\ 2u_2 + u_4 \end{bmatrix} . \text{ avec } \mathbf{d} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

La résolution en norme L_2 et l'application d'un poids diagonal simple donne (ϵ est choisi de manière à ce que l'énergie des résidus soit égale à 1%):

$$[\mathbf{L}^*\mathbf{L} + \epsilon\mathbf{I}]^{-1}\mathbf{L}^*\mathbf{d} = \begin{bmatrix} 0.72 \\ 0.36 \\ 0.36 \\ 0.18 \end{bmatrix}$$

La solution est répartie de manière préférentielle dans le sous-vecteur (u_1, u_2) . Si on utilise une matrice de covariance pour favoriser encore plus la concentration de l'énergie dans (u_1, u_2) (dans l'esprit de l'application d'une norme L_p), sans tenir compte des corrélations entre échantillons voisins, on obtient (les variances sont estimées égales aux carrés des éléments de la solution précédente):

$$\mathbf{C}_{\mathbf{u}} = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.15 & 0 & 0 \\ 0 & 0 & 0.15 & 0 \\ 0 & 0 & 0 & 0.04 \end{bmatrix} \quad \left[\mathbf{L}^*\mathbf{L} + \epsilon\mathbf{C}_{\mathbf{u}}^{-1} \right]^{-1}\mathbf{L}^*\mathbf{d} = \begin{bmatrix} 0.87 \\ 0.38 \\ 0.13 \\ 0.05 \end{bmatrix}$$



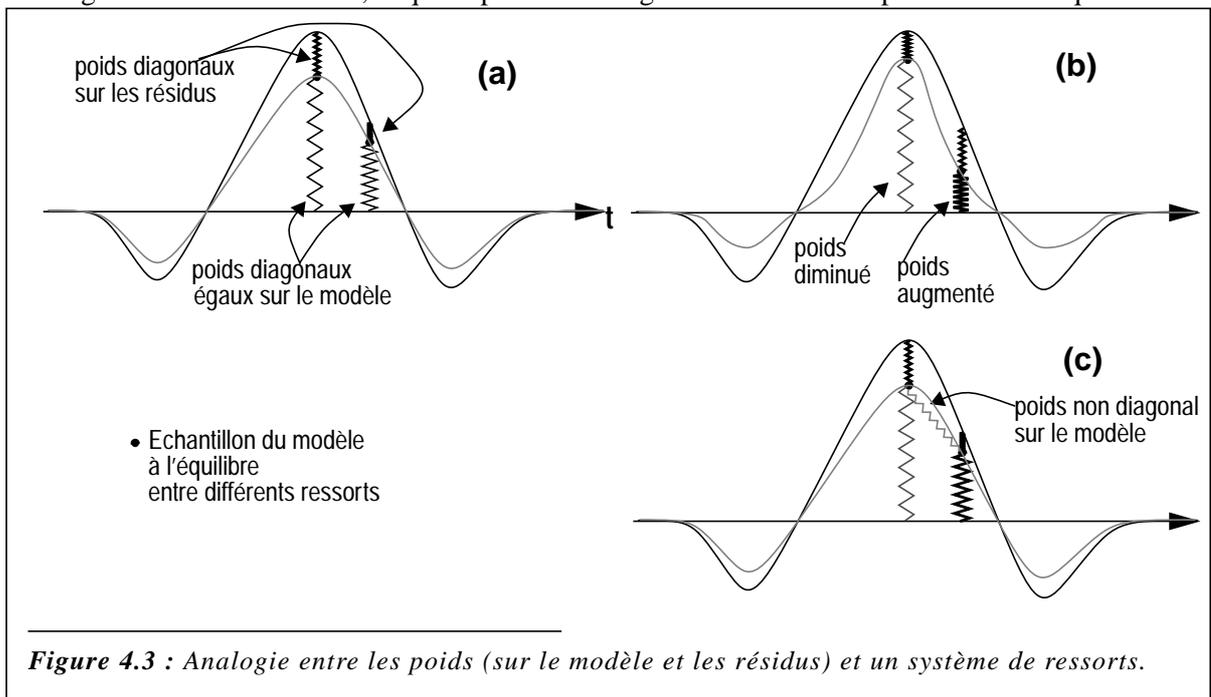
Le sous-vecteur (u1,u2) est bien favorisé, mais on a introduit une distorsion entre les échantillons u1 et u2 (de même entre u3 et u4): le rapport n'est plus égal à 2, car le poids appliqué à u2 est supérieur à celui appliqué à u1. On introduit donc des coefficients non diagonaux qui traduisent que u1 et u2 sont fortement liés:

$$\tilde{\mathbf{C}}_{\mathbf{u}} = \begin{bmatrix} 0.5 & 0.25 & 0 & 0 \\ 0.25 & 0.15 & 0 & 0 \\ 0 & 0 & 0.15 & 0.07 \\ 0 & 0 & 0.07 & 0.04 \end{bmatrix} \quad \left[\tilde{\mathbf{L}}^* \tilde{\mathbf{L}} + \varepsilon \tilde{\mathbf{C}}_{\mathbf{u}}^{-1} \right]^{-1} \tilde{\mathbf{L}}^* \mathbf{d} = \begin{bmatrix} 0.83 \\ 0.43 \\ 0.12 \\ 0.06 \end{bmatrix}$$

On obtient les deux effets souhaités: favoriser la partie (u1,u2) de la solution, et respecter les rapports (u1/u2=u3/u4=d1/d2). Dans la pratique, introduire des poids non diagonaux n'est pas envisageable pour des raisons de coût (mémoire et surtout CPU). Aussi, pour obtenir le même effet avec un poids diagonal, il faut garder le poids constant (ou avec des variations lisses) le long des échantillons dont on veut respecter les rapports d'amplitudes (Thorson and Claerbout, 1985):

$$\tilde{\mathbf{C}}_{\mathbf{u}} = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0.15 & 0 \\ 0 & 0 & 0 & 0.15 \end{bmatrix} \quad \left[\tilde{\mathbf{L}}^* \tilde{\mathbf{L}} + \varepsilon \tilde{\mathbf{C}}_{\mathbf{u}}^{-1} \right]^{-1} \tilde{\mathbf{L}}^* \mathbf{d} = \begin{bmatrix} 0.83 \\ 0.42 \\ 0.12 \\ 0.06 \end{bmatrix}$$

Les poids diagonaux peuvent être envisagés comme des ressorts (dont la raideur est proportionnelle au poids) qui tirent les échantillons du modèle vers 0, contrebalancés par des ressorts qui tirent les échantillons du résidu vers 0 (poids sur les données). Un terme non diagonal est un ressort entre deux échantillons (fig. 4.3). Avec cette analogie, on comprend que l'application de poids différents le long d'un signal déforme ce dernier, et qu'un poids non diagonal est nécessaire pour rétablir l'équilibre.

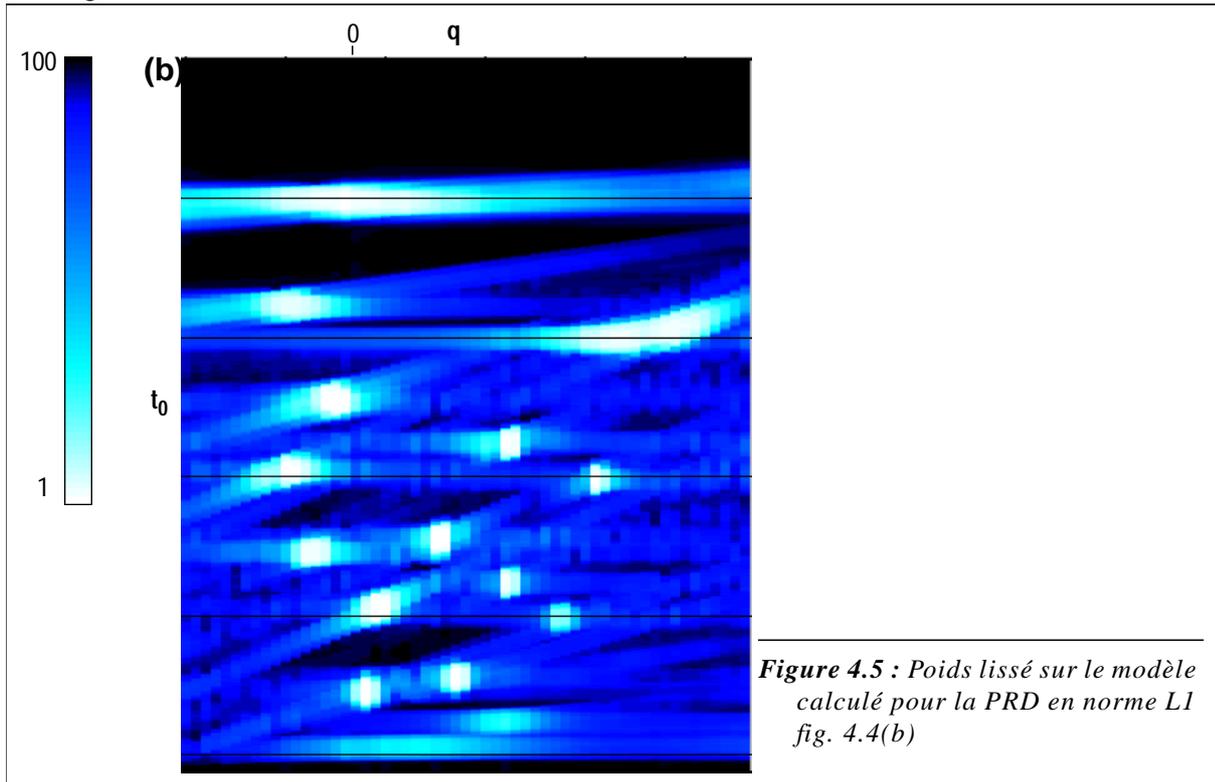


4.2 - Application d'une Norme L_p sur le modèle.

Le poids pour la norme L_p n'est donc plus calculé directement à partir du modèle (équ. (E4.7)), mais à partir de l'enveloppe du modèle, que l'on approxime par une moyenne glissante le long de l'axe du temps, sur la longueur T de l'ondelette sismique (que l'on cherche à préserver):

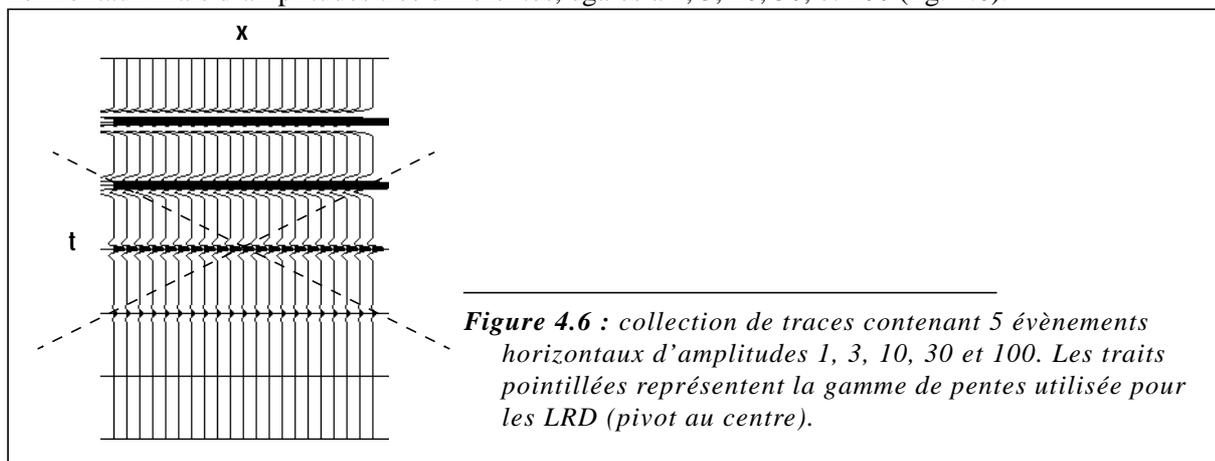
$$\left(\tilde{\mathbf{P}}_{L_p}(\mathbf{u})\right)_{ii} = \frac{1}{(\bar{\mathbf{u}}_i + \eta)^{2-p}} \quad \bar{\mathbf{u}} = \left(\frac{1}{T} \int_{t_i-T}^{t_i+T} \mathbf{u}(t_0, x_0)^2 dt_0 \right)^{\frac{1}{2}} \quad (\text{E4.13})$$

La comparaison de la PRD en norme $L1$ avec et sans lissage du poids (fig. 4.4) montre que la forme de l'ondelette est mieux respectée avec le lissage. Sur la figure 4.5 se trouve le poids lissé, une fois la convergence atteinte.



4.2.5 - Convergence - réduction du nombre d'itérations

On étudie les problèmes de convergence sur une collection de traces ne contenant que des évènements horizontaux mais d'amplitudes très différentes, égales à 1, 3, 10, 30, et 100 (fig. 4.6).



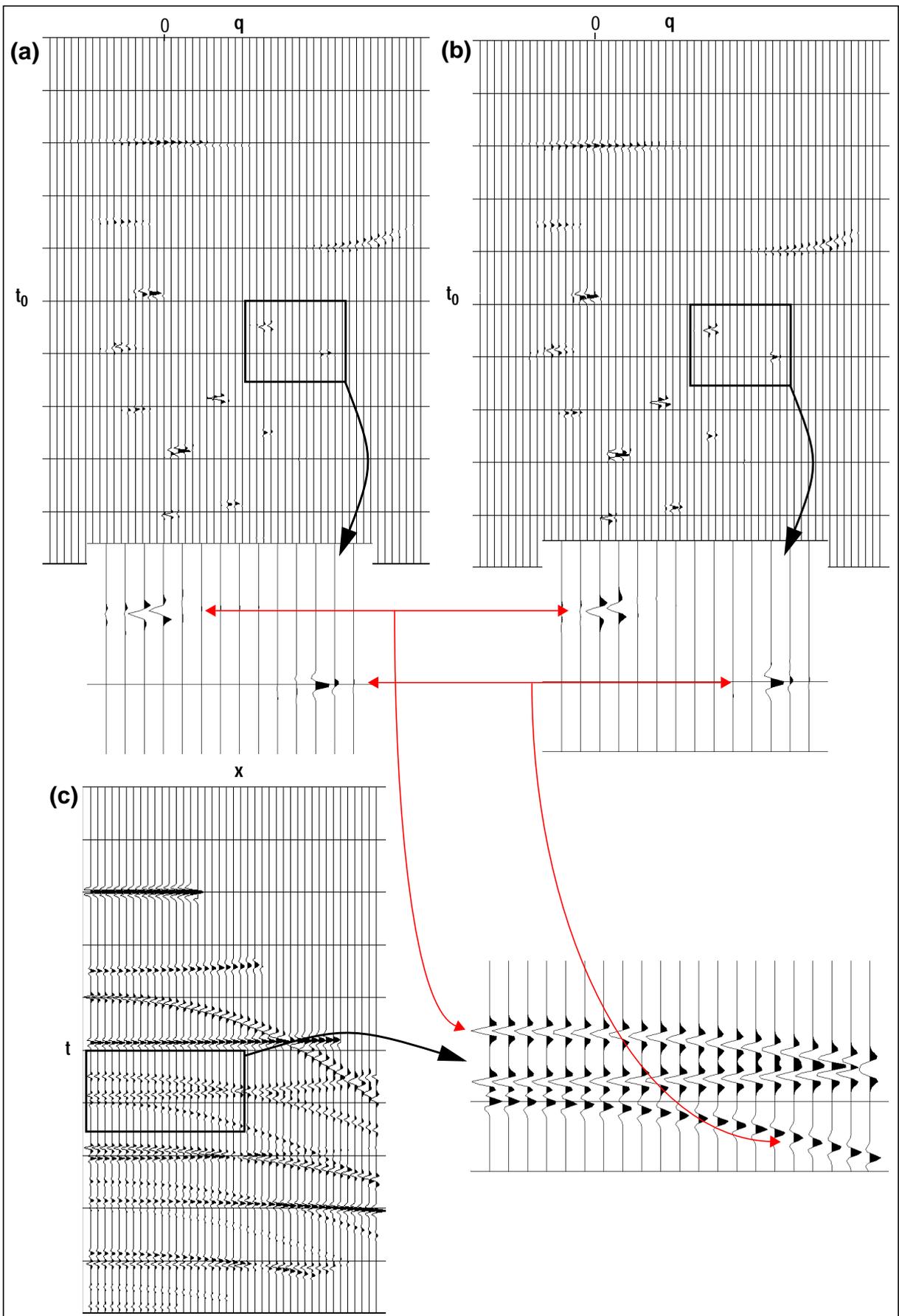
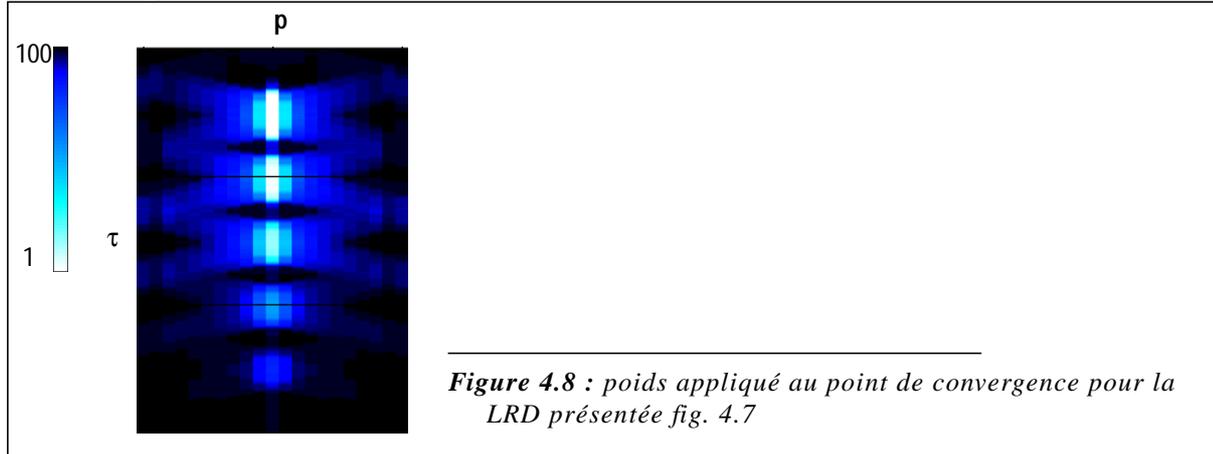


Figure 4.4 : PRDs en norme $L1$ sur le modèle des données fig. 4.2a ($q_{min}=-100$ ms/km², $q_{max}=250$ ms/km², $\Delta q=6.25$ ms/km²). (a) sans lissage du poids (idem 4.2d); (b) avec lissage; (c) rappel des données pour comparaisons sur la forme du signal sismique.

4.2 -Application d'une Norme L_p sur le modèle.

On effectue une LRD en norme L_1 sur ces données (on ajuste ε pour atteindre 1% d'énergie résiduelle, et la dynamique du poids est limitée à 100). Sur la figure 4.7 sont tracés les modèles obtenus au bout de 1, 5 10, et 20 itérations (avec trois gains différents 1, 10 et 100, afin de pouvoir bien visualiser les évènements d'amplitudes très différentes). Sur les premières itérations le modèle prend la forme générale de la solution en norme L_2 (le poids étant égal à l'identité), avec un étalement important de l'énergie autour de la trace $p=0$. Puis au fur et à mesure des itérations, il se rapproche de la solution en norme L_1 , l'énergie étalée se reportant progressivement sur les pics centraux. A l'itération 20, les gros évènements ne sont plus représentés que sur la trace $p=0$, mais il subsiste un certain étalement pour les plus petits (pour des amplitudes inférieures à η , le poids est quasi-constant (équ. (E4.13)), conduisant à traiter cette partie du modèle comme en norme L_2).



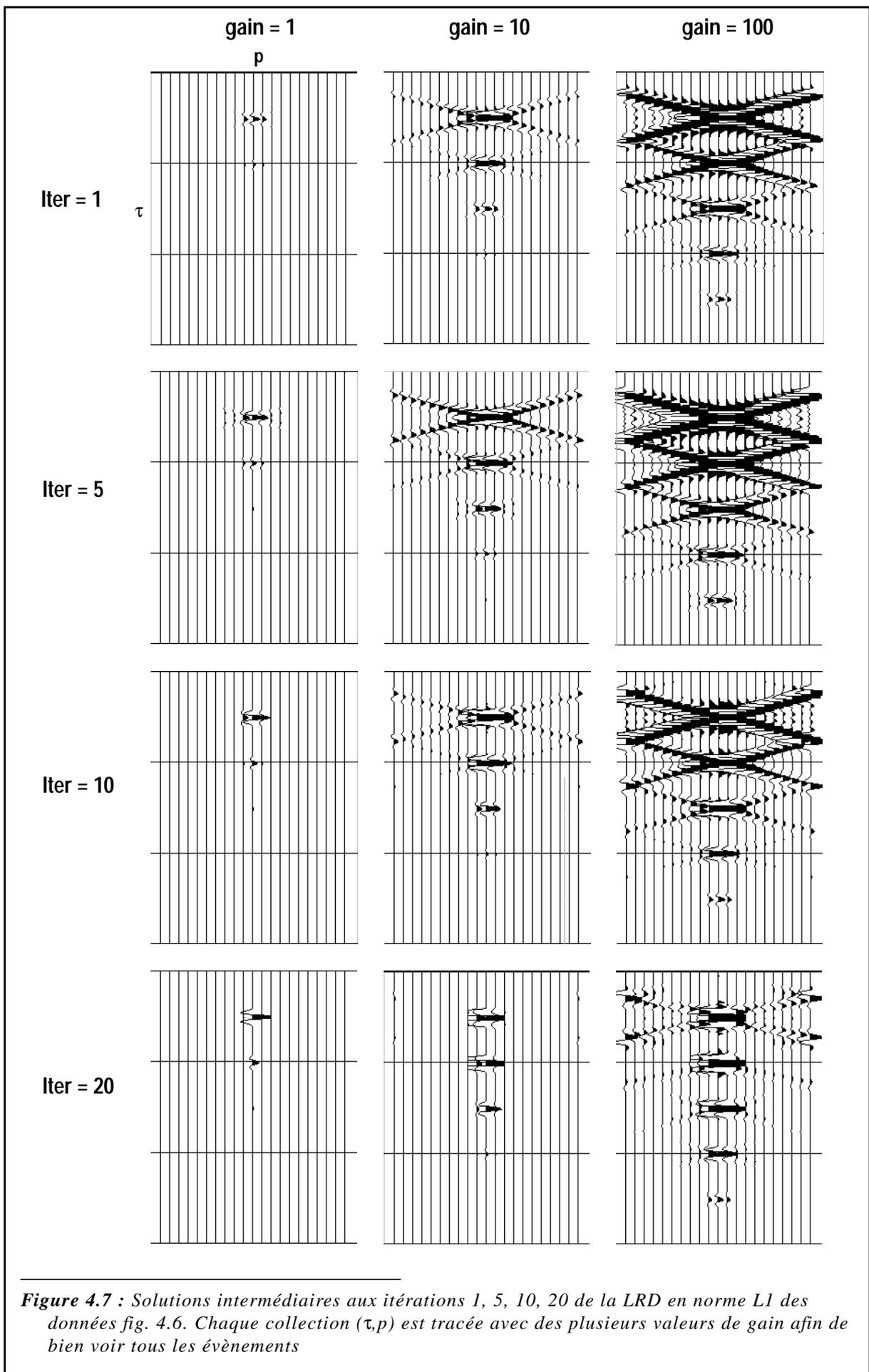
Les courbes de convergences (fig. 4.9) montrent que les résidus diminuent rapidement au cours des premières itérations, et qu'ensuite la norme L_1 du modèle diminue jusqu'au point de convergence, vers l'itération 20 (les petites oscillations autour du point d'équilibre sont dues à la mise à jour de ε pendant les itérations). La même expérience avec un facteur de blanchiment plus faible (0.1% de résidus souhaités) montre un ralentissement notable de la convergence, le modèle notamment tendant très lentement vers son minimum de norme L_1 .

La convergence est donc très affectée par la valeur de ε dans (E4.5). Si les données sont peu bruitées, il faut en principe choisir une valeur très faible. A la première itération le poids est égal à l'identité, conduisant ainsi à un premier modèle $\mathbf{u}(1)$ proche de la solution en norme L_2 , très éloignée de la solution en norme L_1 souhaitée. Lors des itérations suivantes le gradient (équ. (E4.12)) est égal à $\mathbf{P}_{L_p}^{-1} \mathbf{L}^* \mathbf{r} - \varepsilon \mathbf{u}$: le premier terme tend à diriger la solution vers le minimum de résidu, tandis que le second 'remet en cause' une partie du modèle déjà calculé. Si ε est très faible le premier terme est prépondérant; or l'apparition de caractéristiques imposées par la norme L_1 ne peut se faire que sur la partie remise en cause à chaque itération par le second terme, d'où une convergence très lente.

Il est gênant que la convergence dépende à ce point de ε (donc de l'estimation du bruit). Une solution serait d'arrêter les itérations avant le point de convergence, dès lors que l'énergie résiduelle est jugée suffisamment faible. On obtiendrait ainsi une solution intermédiaire entre norme L_2 et norme L_1 . Ce procédé semble néanmoins risqué, car au cours des itérations on n'a aucun contrôle sur les résidus, une énergie résiduelle faible ne signifiant pas nécessairement que toutes les données sont également ajustées.

Si par contre on utilise le premier type d'algorithme IRLS (paragr. 3.5.3), on peut stopper le processus à la fin de n'importe laquelle des décompositions. La norme L_1 de la solution est alors d'autant plus faible que l'on effectue plus de décompositions successives (fig. 4.10). Le point de départ de chaque décomposition doit évidemment être le modèle nul, et non pas la solution de la décomposition précédente, faute de quoi on retombe sur les mêmes problèmes de convergence.

Les courbes de convergence pour une décomposition à poids constant sont superposées aux précédentes sur la figure 4.9 (pour la 3^{ème} décomposition, la 1^{ère} étant celle en norme L_2). La convergence est beaucoup moins affectée par ε qu'avec l'algorithme de type 2. Les évènements de fortes amplitudes se forment dans les premières itérations et changent peu par la suite, les évènements



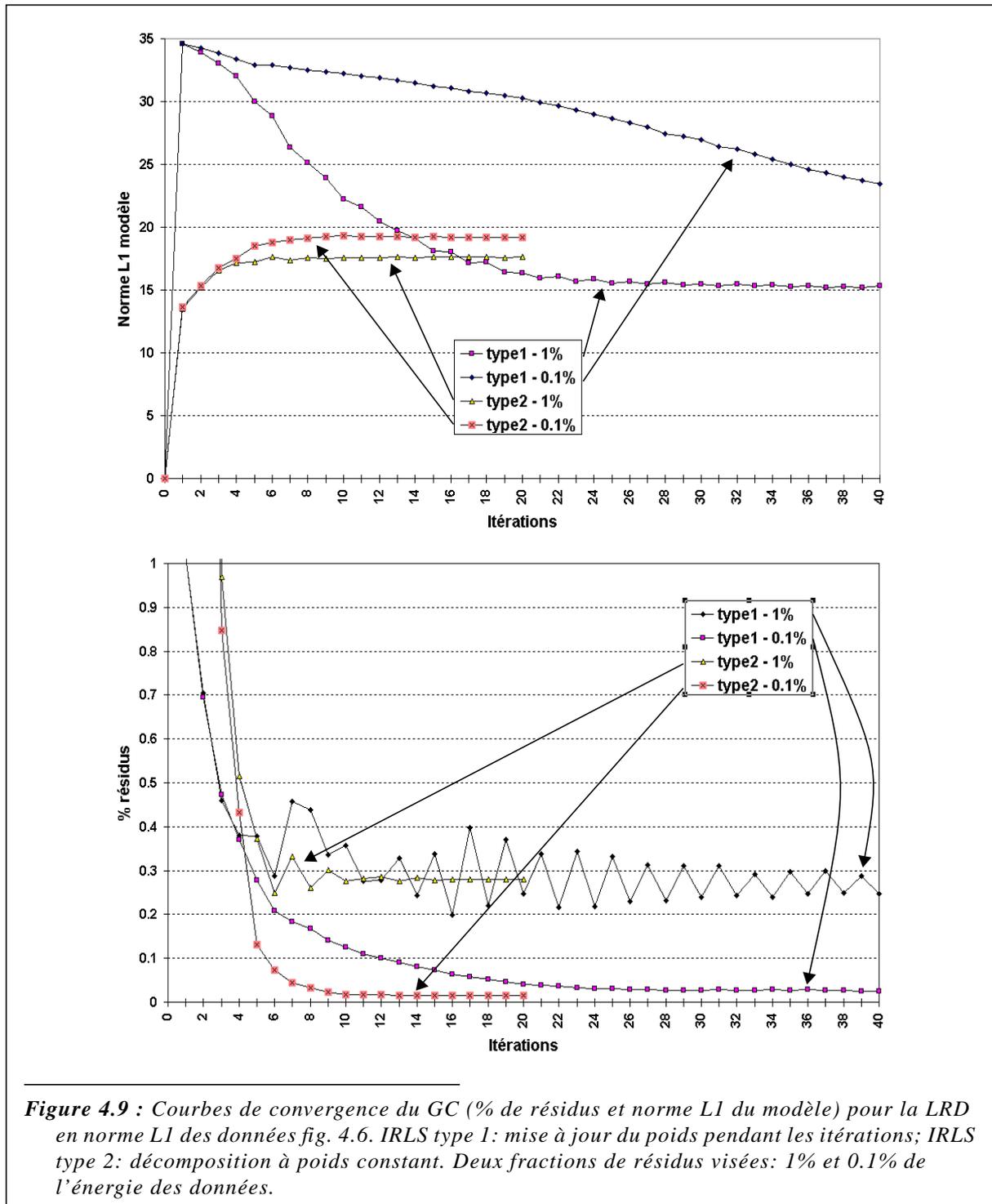
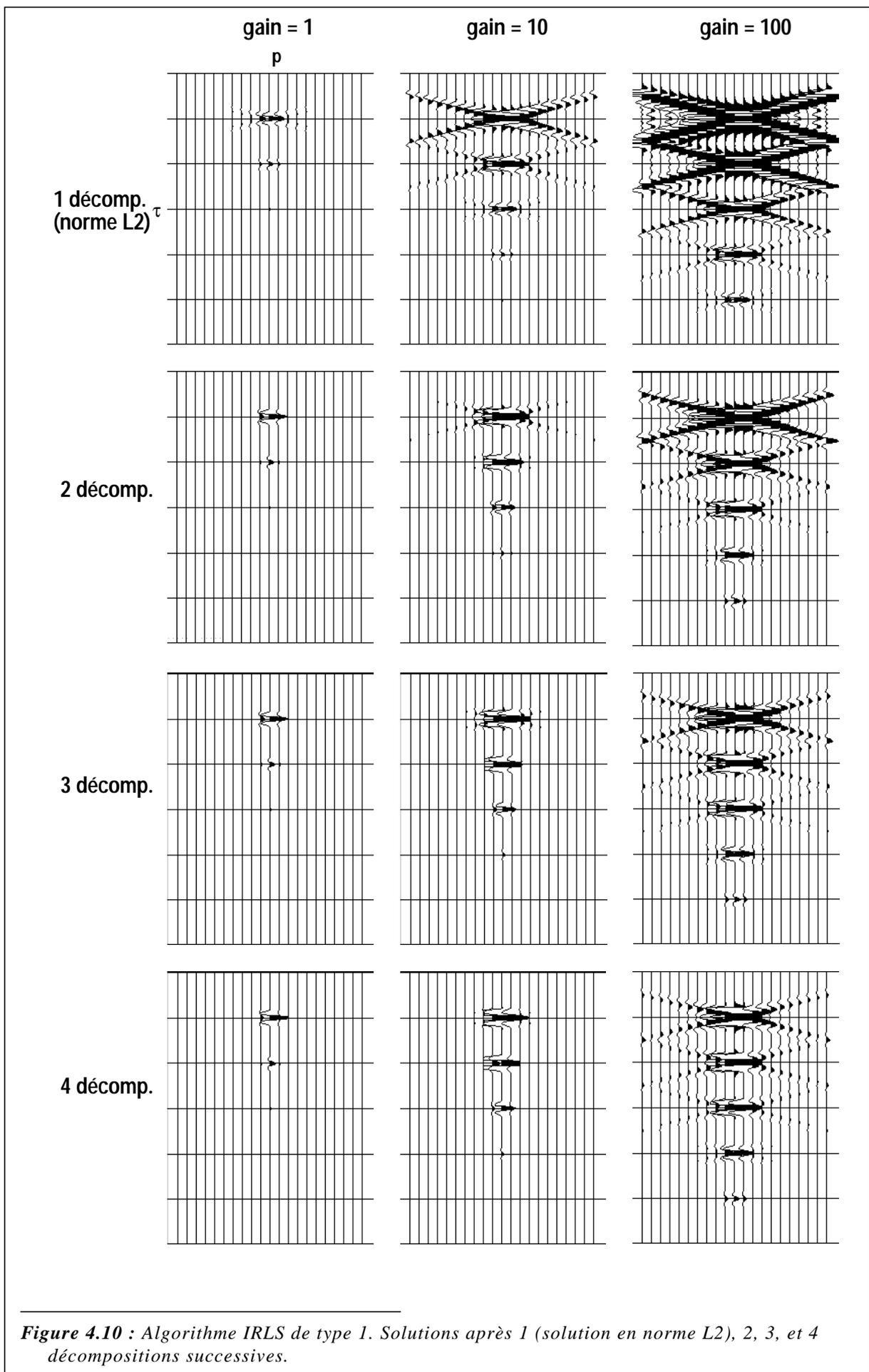
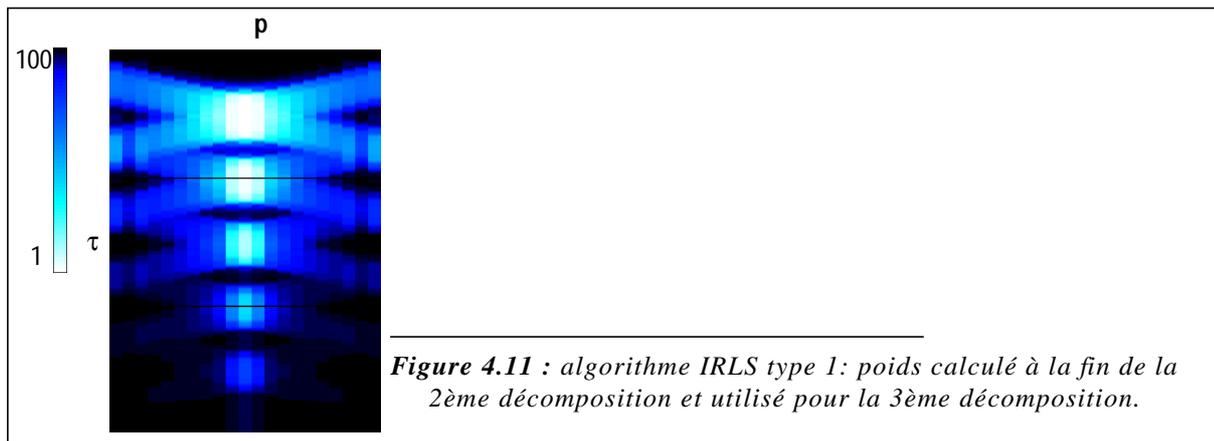


Figure 4.9 : Courbes de convergence du GC (% de résidus et norme L1 du modèle) pour la LRD en norme L1 des données fig. 4.6. IRLS type 1: mise à jour du poids pendant les itérations; IRLS type 2: décomposition à poids constant. Deux fractions de résidus visées: 1% et 0.1% de l'énergie des données.

d'amplitudes de plus en plus faibles se formant au fur et à mesure des itérations (fig. 4.12). Cela résulte du préconditionnement par l'inverse du poids (équ. (E4.12)), qui réduit l'étalement des événements dans le gradient, mais qui pénalise les événements de faibles amplitudes, affectés d'un poids fort. Dans ce schéma, il est donc important d'aller jusqu'au point de convergence pour chaque décomposition, sous peine de ne pas représenter les petits événements. Il est clair également que ce point de convergence est d'autant plus long à atteindre que la dynamique du poids (équ. (E4.10)) est grande.

Le résultat obtenu après 3 décompositions (fig. 4.10), voire même après 2 décompositions, est déjà satisfaisant: la majeure partie de l'énergie étalée a été atténuée au profit des pics centraux.





4.2.6 - Focalisation des évènements - propriétés anti aliasing spatial

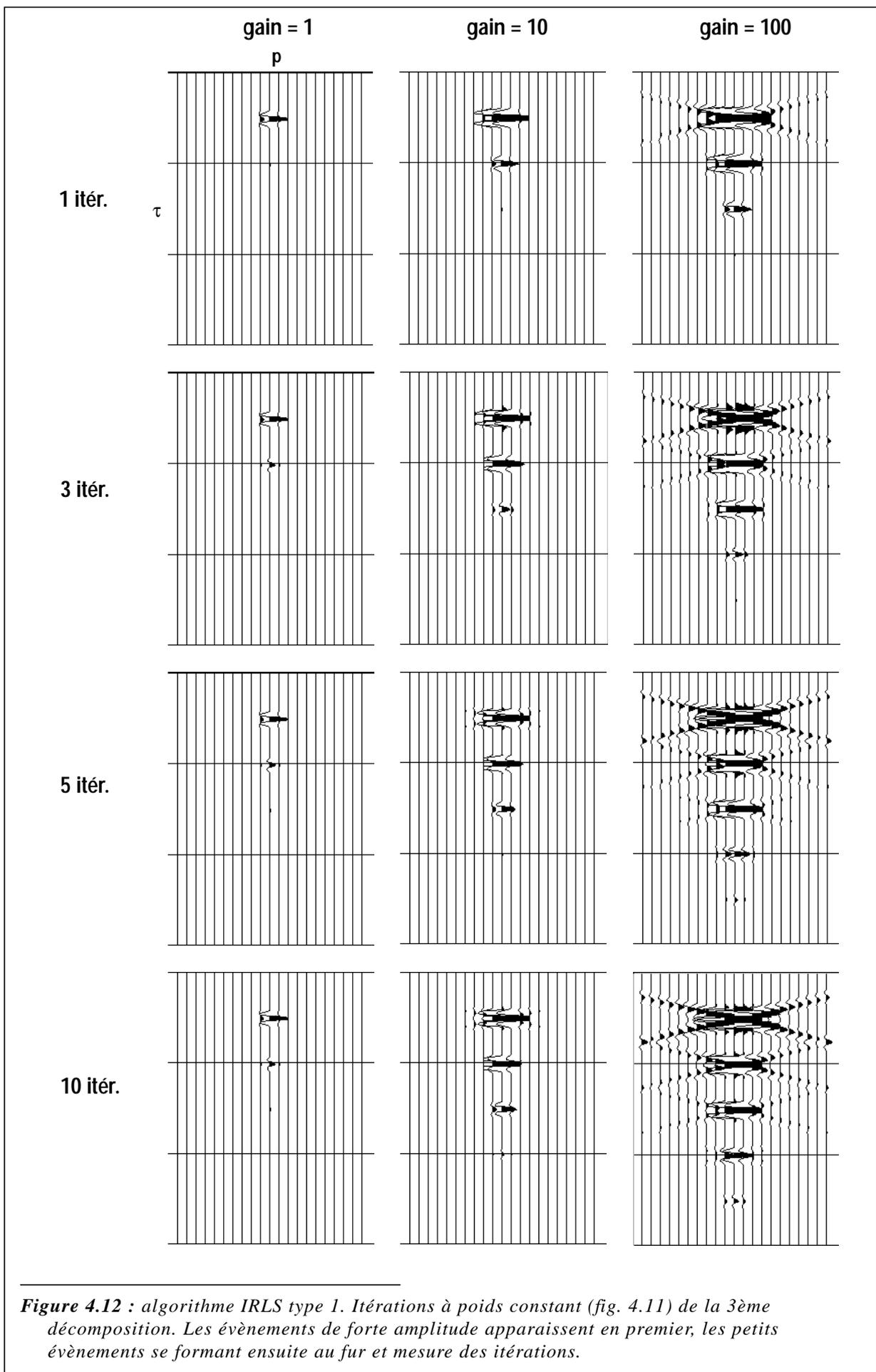
La propriété la plus intéressante des solutions en norme L_p ($p < 2$) est de favoriser la concentration de l'énergie en des points privilégiés du modèle, permettant une meilleure focalisation des évènements que la norme L_2 . La propriété anti aliasing découle de cette propriété. L'aliasing se traduit en effet par une dispersion de l'énergie dans le modèle (fig. 2.14 et 2.19). Une décomposition parcimonieuse permet donc de ramener cette énergie sur le pic principal de l'évènement.

On atteint néanmoins une limite de ce principe de meilleure focalisation pour des évènements qui par nature ne peuvent pas être représentés en seul point du modèle: une hyperbole, une droite d'amplitude non constante, ne peuvent pas se réduire à un seul point dans une LRD. Les décompositions de figure 4.2 montrent que l'évènement très courbé situé à 400ms, qui s'écarte du modèle parabolique de la PRD, reste moins focalisé que les autres dans la PRD en norme L_1 .

Une des applications les plus intéressantes de cette focalisation est le traitement anti-multiples par séparation dans le domaine de la transformée parabolique des CMP après correction NMO (ou dans le domaine hyperbolique avant correction NMO) (Hampson, 1987; Canadas et. al, 1991; Foster And Mosher, 1991). Ce traitement repose sur la différence de NMO éventuel entre les multiples et les primaires, et est donc a priori plus adapté aux multiples longues périodes. Lorsque la différence de NMO est suffisamment importante, une PRD en norme L_2 est suffisante. Mais lorsqu'elle est faible (en général pour les temps importants, ou pour des multiples à courte période), l'étalement des évènements dans le modèle empêche d'une part un pointé de séparation précis et entraîne d'autre part un mélange des informations (l'énergie des multiples déborde dans le domaine des primaires et réciproquement).

On illustre ceci sur des données contenant 5 évènements horizontaux (les primaires supposés, le premier d'amplitude 10 à 2100ms, les autres d'amplitude 1 à 2200, 2500, 2550 et 2600ms) auxquels se superposent 3 évènements faiblement courbés ($q=50$ et 100 ms/km^2 , les multiples possédant une courbure résiduelle après correction NMO, d'amplitudes 10 à 2300, 2475 et 2550ms) (figure 4.13). On effectue une PRD en norme L_2 (4.13b) et une en norme L_1 (4.13d) de ces données, puis on reconstruit les primaires en ne conservant que le domaine des courbures ($q < 25 \text{ ms/km}^2$). La reconstruction après PRD en norme L_2 (4.13c) laisse apparaître beaucoup de 'résidus' de multiples, dus aux traînées dans la PRD provenant des multiples et débordant dans la partie ($q < 25 \text{ ms/km}^2$). L'utilisation de la norme L_1 dans la PRD résout pratiquement complètement ce problème (4.13e). Il faut néanmoins garder à l'esprit que les conditions sont quasi idéales pour l'application de la norme L_1 : pas de bruit sur les données et évènements parfaits par rapport au modèle parabolique.

On considère maintenant une collection CMP corrigée de NMO (données réelles), entachée de multiples très énergiques mais avec des courbures résiduelles très importante (fig. 4.14). On effectue des



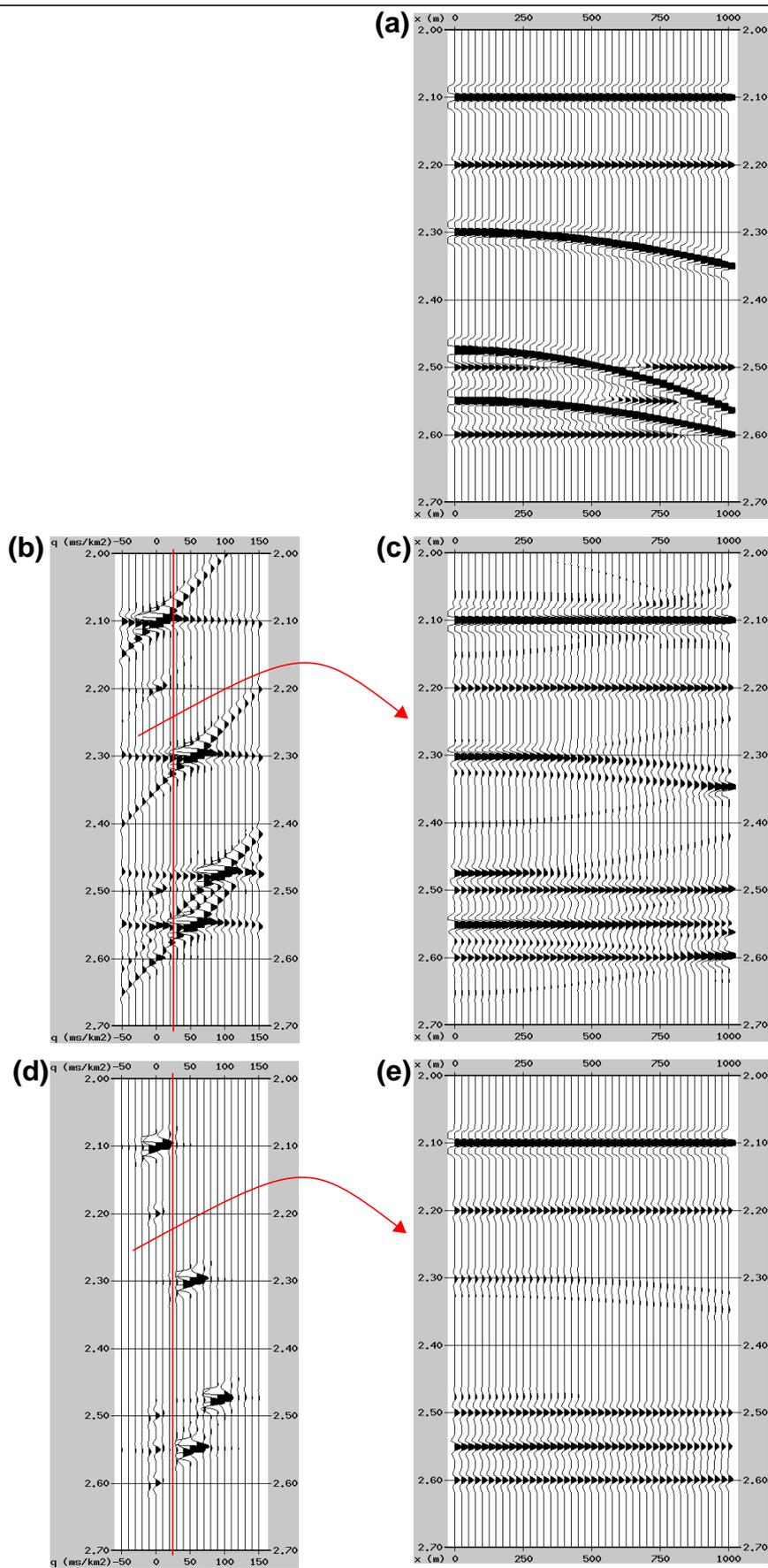


Figure 4.13 : (a) données (évènements d'amplitudes 1 et 10). (b) (d) PRD norme L2 et norme L1.
 (c) (e) Reconstructions avec les domaines $q < 25 \text{ ms/km}^2$

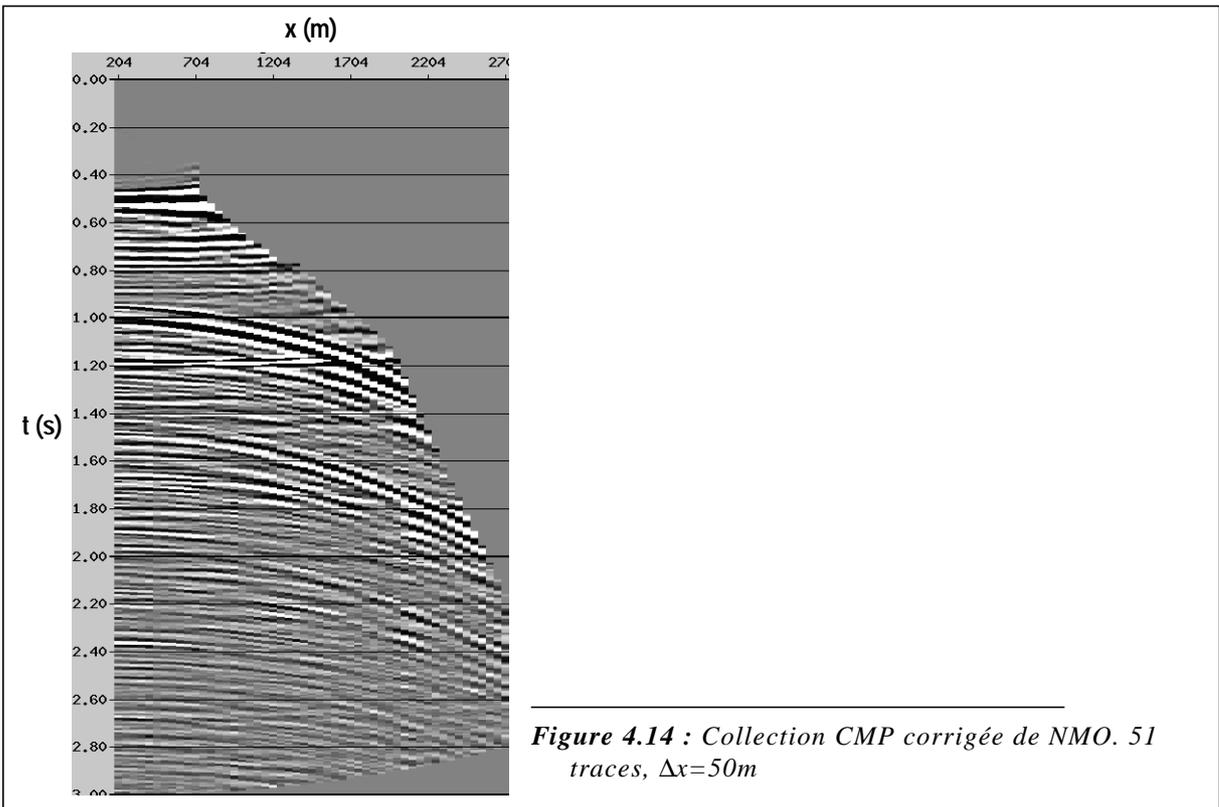


Figure 4.14 : Collection CMP corrigée de NMO. 51 traces, $\Delta x=50m$

PRD de ce CMP (en norme L2 et en norme L1, 1% de résidus visés, IRLS type 1 avec 3 décompositions pour la norme L1), on pointe une loi de séparation entre les réflexions primaires et multiples, puis on reconstruit le domaine des primaires seulement (fig. 4.15). Les événements sont mieux individualisés dans la PRD en norme L1, facilitant grandement le pointé. Le CMP débarrassé des multiples semble également plus propre en utilisant la norme L1.

4.2.7 - Sensibilité au bruit - perte de certaines données

Nous étudions ici le comportement de la décomposition en norme L1 vis à vis du bruit, et donc du réglage du terme de régularisation dans (E4.5).

Dans une décomposition en norme L_p , on suppose que les échantillons du modèle obéissent à une loi gaussienne généralisée (équ. (E4.4)). Du point de vue statistique, et similairement au cas de la norme L2 (équ. (E3.39)) la fonction coût à minimiser doit être:

$$\varphi(\mathbf{u}) = \frac{1}{2\sigma_n^2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|_{L2}^2 + \frac{1}{p\sigma_{up}^p} \|\mathbf{u}\|_{Lp}^p \quad (E4.14)$$

où σ_n est l'écart type du bruit (connue ou fixée arbitrairement à priori), et σ_{up} est la dispersion moyenne dans le modèle pour la statistique gaussienne généralisée d'ordre p.

On devrait donc fixer le terme de régularisation dans (E4.5) égal à:

$$\varepsilon = \frac{2\sigma_n^2}{p\sigma_{up}^p} \quad (E4.15)$$

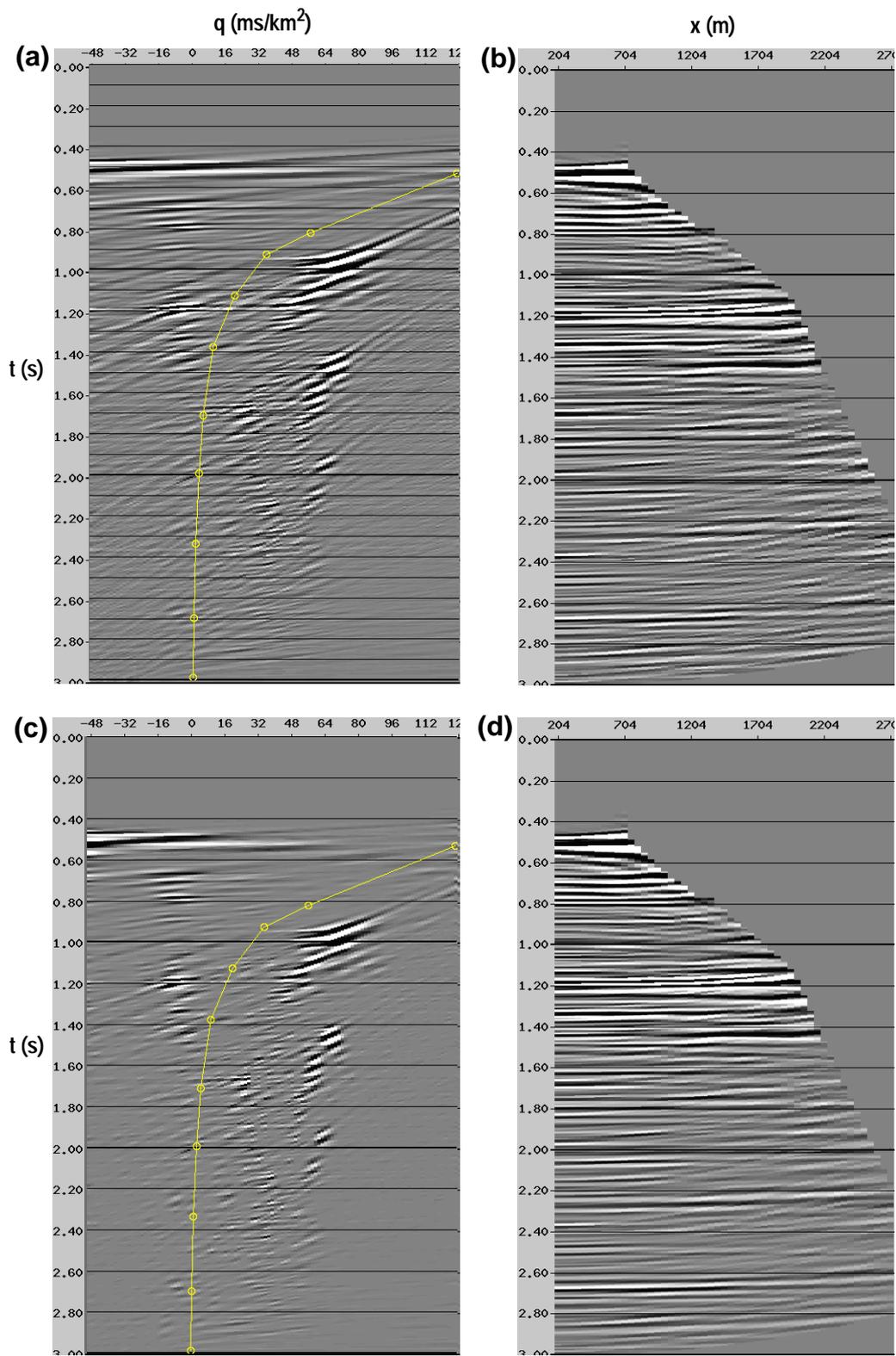


Figure 4.15 : (a) PRD en norme L_2 du CMP fig. 4.14, avec pointé de séparation primaires (à gauche) /multiples (à droite). (b) reconstruction de la partie à gauche du pointé. (c) et (d) Idem avec norme L_1 au lieu de norme L_2 . Les reconstruction (b) et (d) sont égalisées en temps pour les besoins de la visualisation.

Pour cela on estime la dispersion dans le modèle d'après le modèle lui-même au cours des itérations du gradient conjugué:

$$\sigma_{up}^p \leftarrow (\|\mathbf{u}\|_{L_p}^p / n_u) \approx ((\mathbf{u}, \mathbf{P}_{L_p} \mathbf{u}) / n_u) \quad (\text{E4.16})$$

On peut corriger cette estimation d'après la valeur des résidus de la même façon que dans l'équ. (E3.42).

On a vu (paragr. 3.6.2) qu'en norme L2 le facteur d'amortissement est un paramètre peu sensible et qu'il a peu d'influence sur le filtrage du bruit aléatoire. Pour étudier l'influence de la proportion de bruit fixée à priori pour la décomposition en norme L1, on reprend les données de la figure 3.3 sur lesquelles on effectue des LRD en norme L1 (fig. 4.16). Une estimation nettement trop faible du bruit (0.02% des données) conduit à une décomposition totalement bruitée (4.16d), bien que la reconstruction ne contienne pas plus de bruit que les données. Le bruit sur la solution est donc dans le noyau de \underline{L} . En augmentant cette estimation à 0.2% (4.16c) puis 2% (4.16b), le bruit diminue notablement sur la décomposition, mais aussi sur la reconstruction. A 20% estimé, la décomposition et la reconstruction ne contiennent pratiquement plus de bruit. La valeur du terme de régularisation a ici une influence très forte sur le filtrage du bruit aléatoire contenu dans les données, et sur la quantité de bruit présente dans la décomposition. Ces solutions parcimonieuses favorisent les événements cohérents. Sur la reconstruction fig. 4.16a, où le terme de régularisation est très fort, les événements courbes sont moins bien reconstruits que l'évènement linéaire, car ils ne rentrent pas dans le modèle de la LRD (droites d'amplitudes constantes) et sont traités en partie comme du bruit.

Un bon réglage du facteur de régularisation permet donc de filtrer efficacement le bruit sur les données, alors qu'un mauvais réglage conduit à une solution trop bruitée (sous-estimation) ou à l'atténuation non souhaitée d'évènements cohérents (surestimation) (rappelons qu'en norme L2 ce terme a peu d'influence sur le filtrage du bruit, mais qu'en contrepartie un mauvais réglage n'a pas de conséquences dommageables).

On peut reprendre l'exemple de la figure 4.13 en ajoutant du bruit gaussien sur les données, d'amplitude moyenne comparable aux petits évènements horizontaux (fig. 4.17a). La présence de bruit oblige à utiliser un facteur de régularisation plus important dans les PRD (calculé en supposant une proportion de 10% de bruit sur les données). Lorsque l'on passe de la norme L2 (4.17d) à la norme L1.5 (4.17c), puis à la norme L1 (4.17b), on note une meilleure séparation des évènements et une amélioration du rapport S/B, mais aussi une atténuation sensible de l'amplitude des petits évènements, à la fois sur la PRD et sur la reconstruction. Cette atténuation résulte directement de l'application d'un poids élevé (d'autant plus élevé que l'évènement est d'amplitude faible), associé à un facteur de régularisation élevé.

Ce phénomène peut poser problème lorsque l'on travaille sur des données qui possèdent une grande dynamique (avant correction de divergence sphérique par exemple), les évènements faibles se trouvant pénalisés, voire même en dessous de la dynamique du poids. Il faudrait appliquer une égalisation (avec une fenêtre glissante en temps assez longue) sur les données avant la décomposition. Pour éviter cela, on effectue une égalisation uniquement sur le calcul du poids: le poids minimum est déterminé par l'amplitude maximum dans le modèle (équ. (E4.13)). On effectue une égalisation (en temps uniquement) sur le poids de telle sorte que cette valeur de poids minimum se retrouve sur les maximums locaux du modèle (local signifiant dans une fenêtre contenant toutes les traces et d'une longueur en temps à déterminer). On peut voir figure 4.18 le poids brut et avec égalisation calculés pour la PRD en norme L1 de la figure 4.15c.

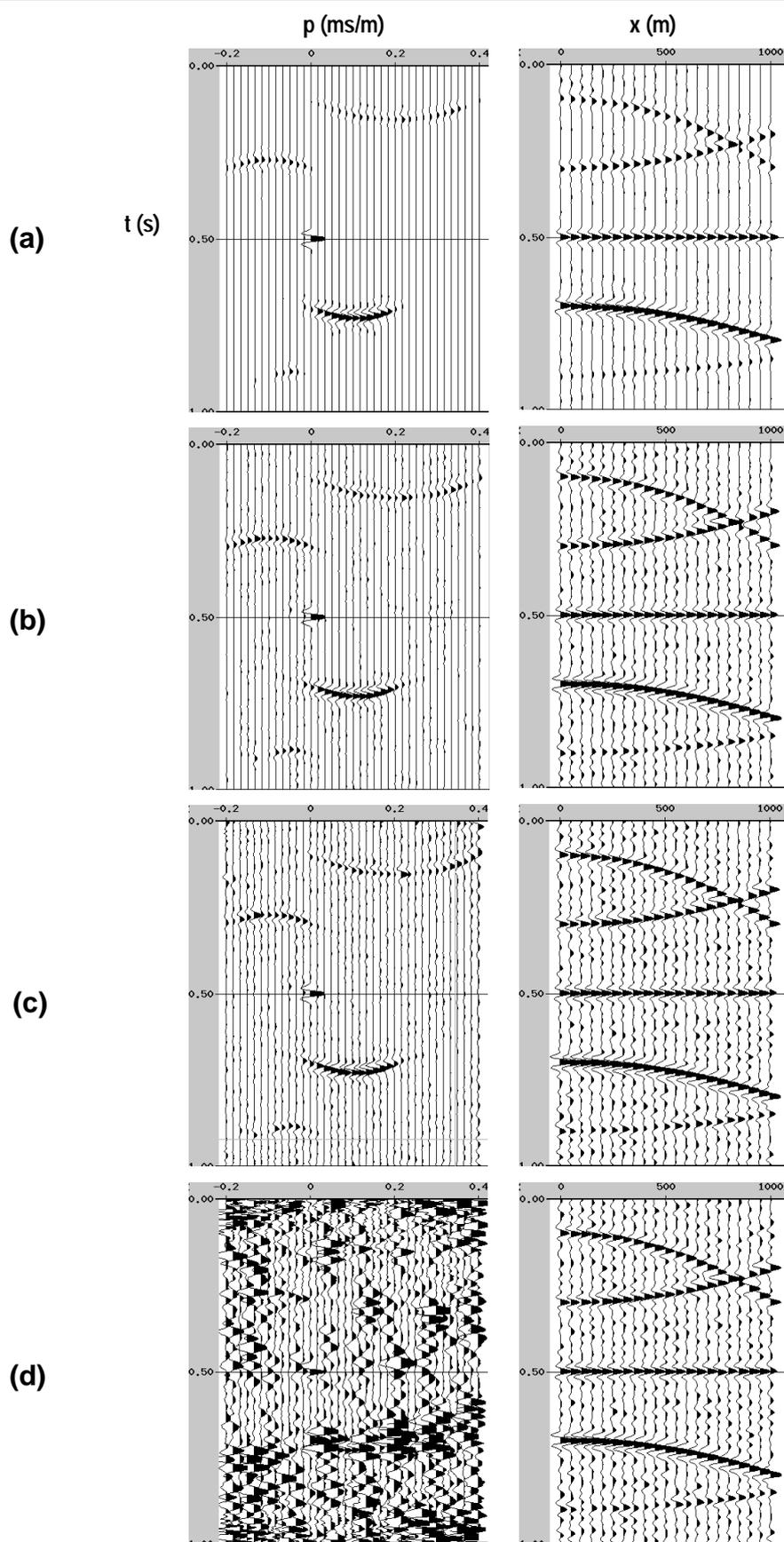


Figure 4.16 : LRDs en norme L_1 (à gauche) des données fig. 3.3 en utilisant plusieurs valeurs de facteur d'amortissement déterminés d'après la proportion d'énergie estimée du bruit sur les données. La reconstructions dans la colonne de droite. Les pentes utilisées s'échelonnent entre -200 et 400 ms/km. (a) 20% de bruit estimé (b) $\epsilon=2\%$ (c) $\epsilon=0.2\%$ (d) $\epsilon=0.02\%$

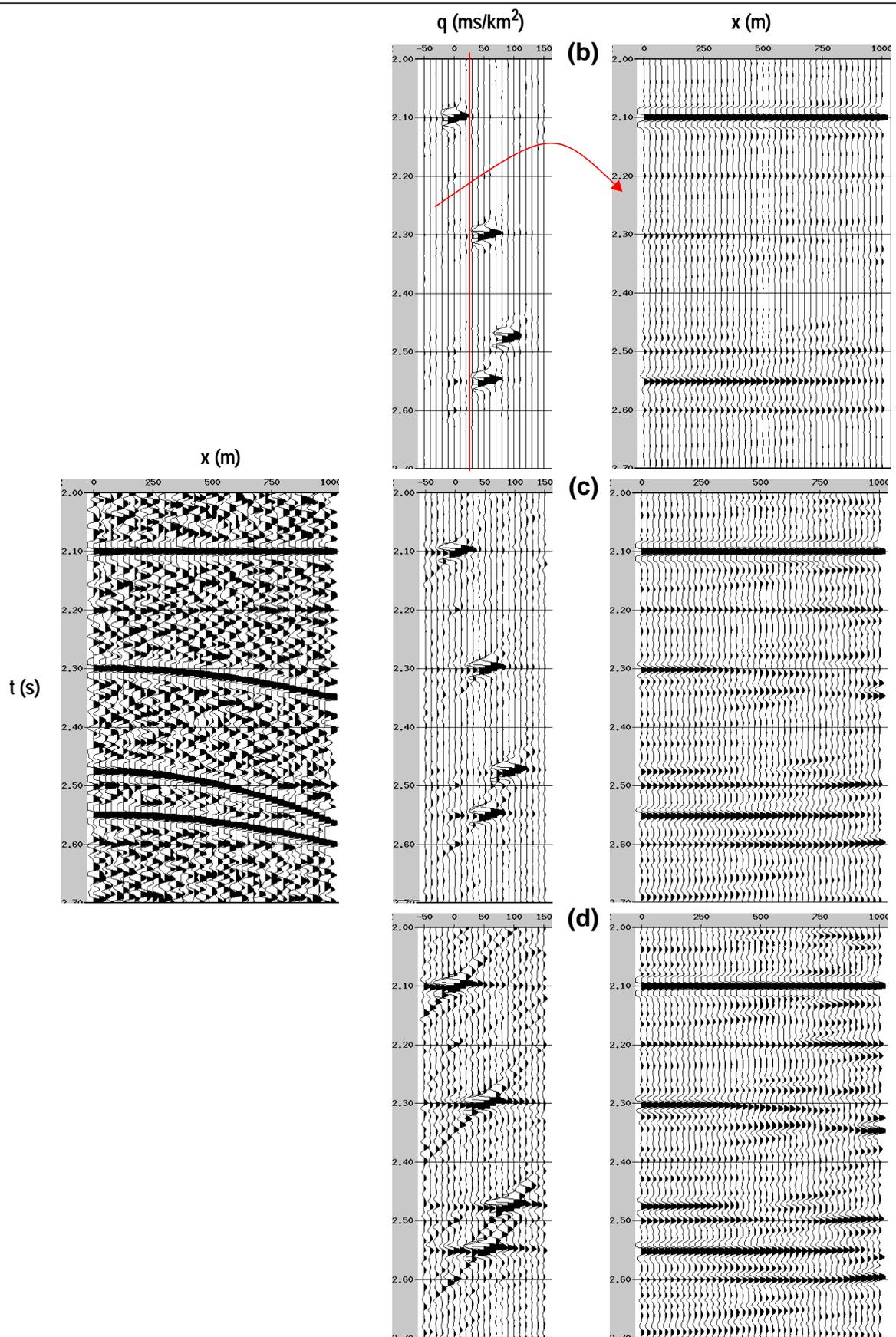
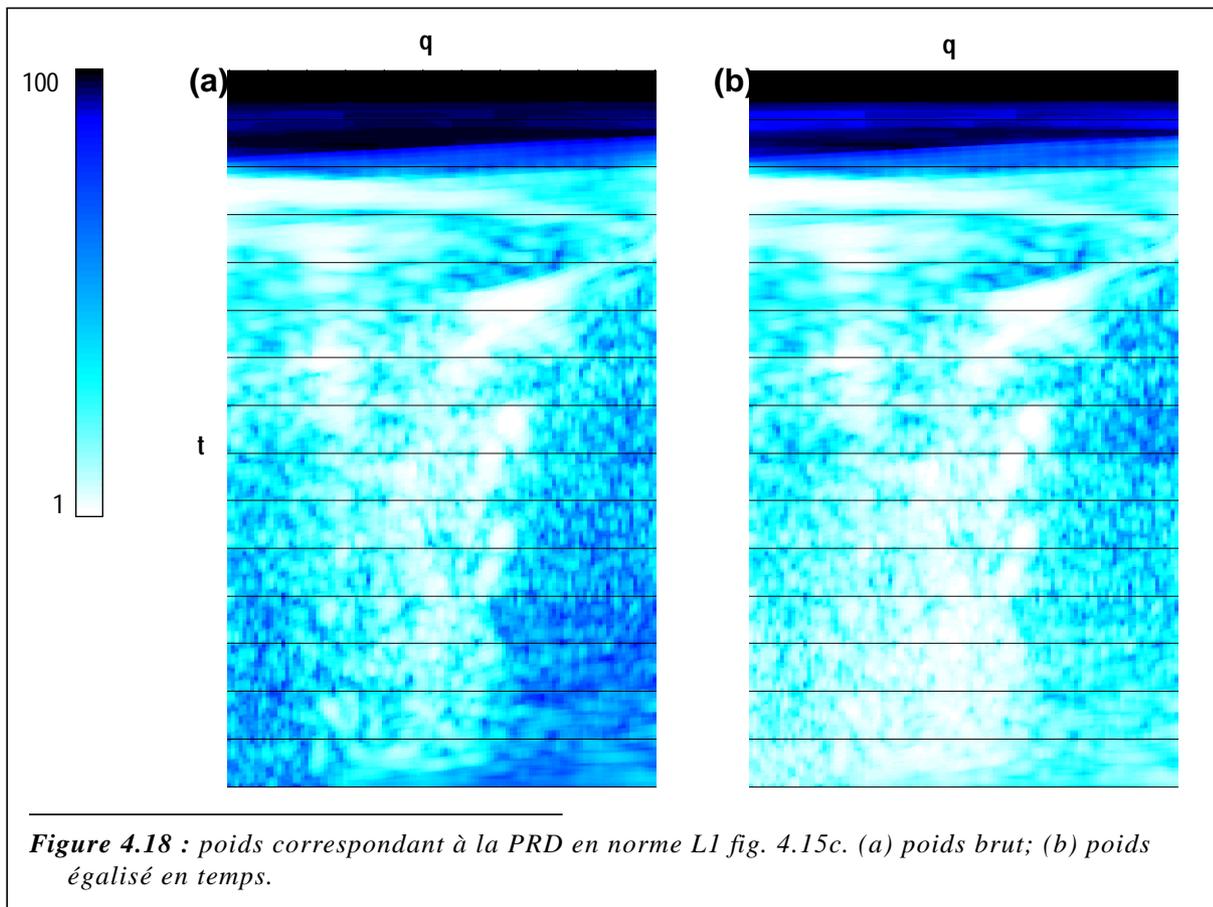


Figure 4.17 : *Gauche:* mêmes données que fig. 4.13a, entachées d'un bruit gaussien (de niveau comparable à l'amplitude des petits évènements). *Milieu:* PRD en norme L1 (b), L1.5 (c) et L2 (d), proportion de résidus estimé 10%. *Droite:* reconstructions des domaine $q < 25\text{ms/km}^2$. Gains=10 par rapport aux plus forts évènements.



4.3 - Application d'une Norme L_p sur les données.

4.3.1 - Présentation

L'application de la norme L_p ($p < 2$) sur le modèle vise à prendre en compte le caractère intrinsèquement parcimonieux des modèles. Lorsque l'on applique, comme cela a été le cas jusqu'ici, la norme L_2 pour la minimisation des résidus, on suppose implicitement que ceux-ci obéissent à une distribution gaussienne. Cette supposition semble raisonnable pour la majorité des bruits aléatoires. Mais les données peuvent également être contaminées par des bruits très énergiques mais très localisés, expliqués (récepteur parasité par une ligne haute tension, ou bruité par un animal) ou non. Ces bruits sont généralement présent sur des traces isolées, s'étalant sur toute une trace ou localisés en temps (auquel cas on les qualifie souvent de 'burst noise'), et n'obéissent pas à une statistique gaussienne, mais plutôt à des distributions plus parcimonieuses. D'où l'idée de les traiter en appliquant une norme L_p ($p < 2$) sur la minimisation des résidus (Nichols, 1994).

On minimise ici:

$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|_{L_p}^p + \frac{\varepsilon}{2} \|\mathbf{u}\|_{L_2}^2 \quad (E4.17)$$

De la même manière que pour l'application de la norme L_p sur le modèle, on approxime localement celle-ci en utilisant un produit scalaire pondéré par un poids diagonal dépendant des résidus (équ. (E4.6), (E4.7), (E4.13)).

Pour exactement les mêmes raisons que pour le poids sur le modèle, ce poids est majoré par l'utilisation d'une petite quantité η (petite devant les valeurs attendues dans les résidus), et lissé sur la

longueur T de l'ondelette sismique (on le calcule sur une approximation de l'enveloppe des résidus), afin de ne pas déformer celle-ci.

$$\left(\mathbf{P}_{Lp} \right)_{ii} = \frac{p}{(\bar{r}_i + \eta)^{2-p}} \quad \bar{r} = \left(\frac{1}{T} \int_{t_i-T}^{t_i+T} \mathbf{r}(t, x)^2 dt \right)^{\frac{1}{2}} \quad (E4.18)$$

Le problème à résoudre est maintenant:

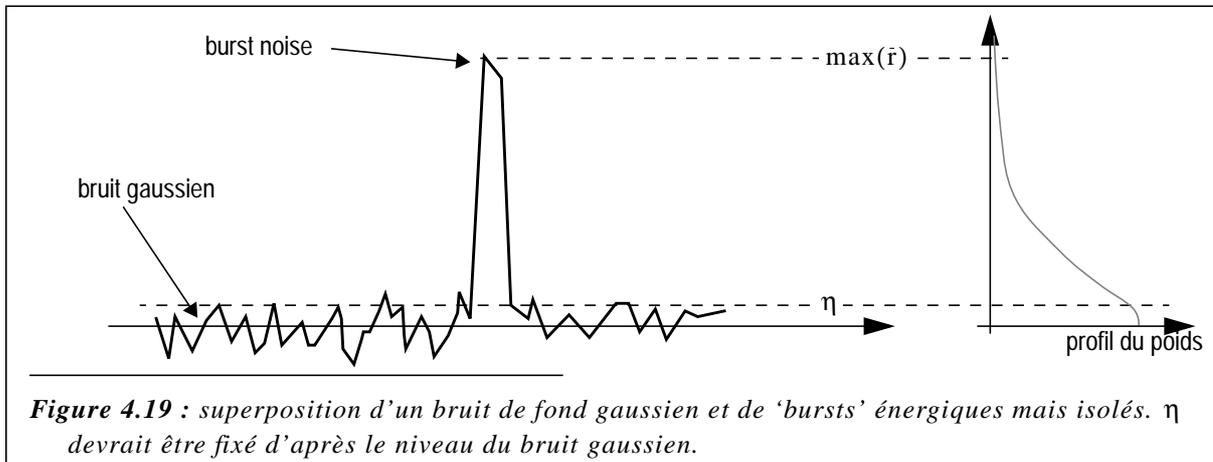
$$\psi(\mathbf{u}) = \frac{1}{2}(\mathbf{d} - \mathbf{L}\mathbf{u}, \mathbf{P}_{Lp}(\mathbf{r})(\mathbf{d} - \mathbf{L}\mathbf{u})) + \frac{\varepsilon}{2}\|\mathbf{u}\|^2 \quad (E4.19)$$

Un choix typique pour η est:

$$\eta = \frac{\max(\bar{r})}{\lambda} = \frac{\max(\bar{r})}{100} \quad (E4.20)$$

On sait que le bruit d'amplitude inférieure à η est traité comme en norme L2 (domaine de poids quasi constant), donc comme du bruit gaussien. Souvent les données contiennent un bruit de fond plus ou moins gaussien, auquel peuvent se superposer des bruits forts et isolés. η devrait être fixé relativement au niveau de ces deux types de bruit (fig. 4.19). Dans cette optique, il devrait également être fixé une fois pour toute et ne pas varier au cours des itérations de l'algorithme IRLS. On le détermine donc à partir des données, sans le remettre à jour ensuite:

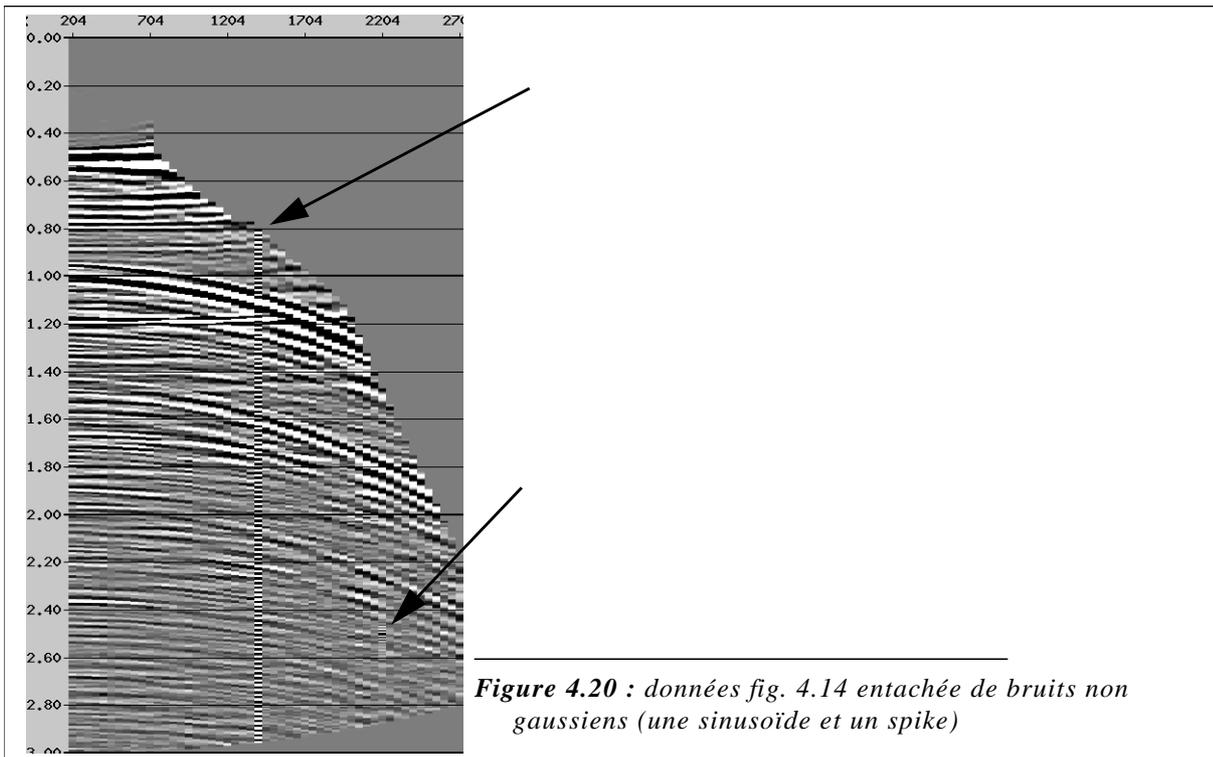
$$\eta = \frac{\max(\bar{\mathbf{d}})}{\lambda} \quad (E4.21)$$



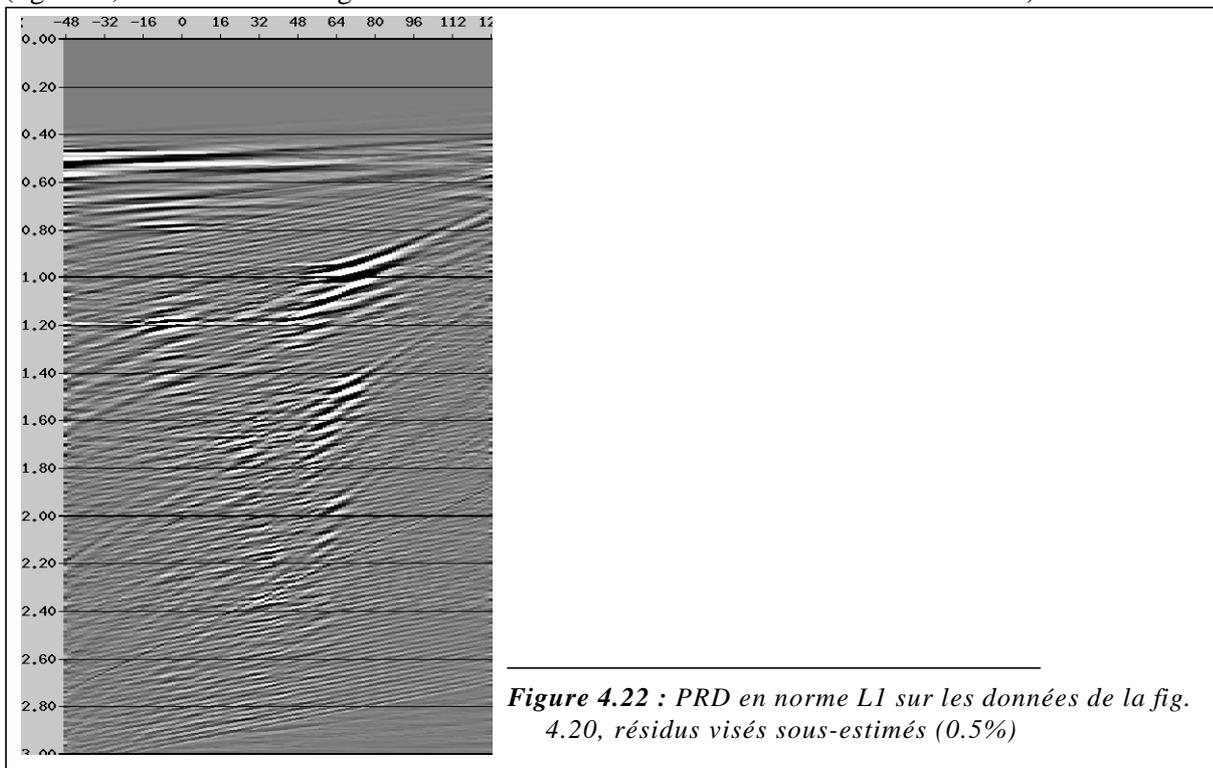
On utilise pour la résolution l'algorithme IRLS de type 1, en résolvant plusieurs fois de suite le problème à poids constant, poids remis à jour entre chaque décomposition. A la différence de ce que l'on fait en norme L_p sur le modèle, on peut calculer un poids dès la première décomposition à partir des données elles-mêmes (qui constituent en fait le premier résidu disponible).

4.3.2 - Meilleurs traitement des bruits non gaussiens

Afin de tester la norme L_p sur les données, on reprend la collection CMP après NMO de la figure 4.14 en ajoutant des bruits localisés: une sinusoïde à 50Hz sur la trace 26, et un spike sur la trace 41 à 2500ms. On effectue (de la même façon qu'en figure 4.15) des PRD suivies des reconstructions du domaine des primaires. La PRD en norme L_2 classique (fig. 4.21a) est fortement contaminée par le bruit induit par le



spike et la sinusoïde, que l'on retrouve d'ailleurs en grande partie sur la reconstruction. Ce bruit induit peut gêner le pointé des événements dans la PRD. L'utilisation de la norme L_1 sur les données (fig. 4.21c) réduit considérablement le bruit, qui est pratiquement éliminé sur la reconstruction. Les deux PRD ont été calculées en visant 5% de résidus. Pour la norme L_1 , on effectue deux décompositions successives (20 itérations chacune). Ce résultat intéressant sur le bruit doit cependant être nuancé: si on vise une quantité de résidus trop faible, le bruit contamine le modèle de la même façon qu'en norme L_2 (fig. 4.22, idem PRD de la fig. 4.21c mais en estimant les résidus à 0.5% au lieu de 5%).



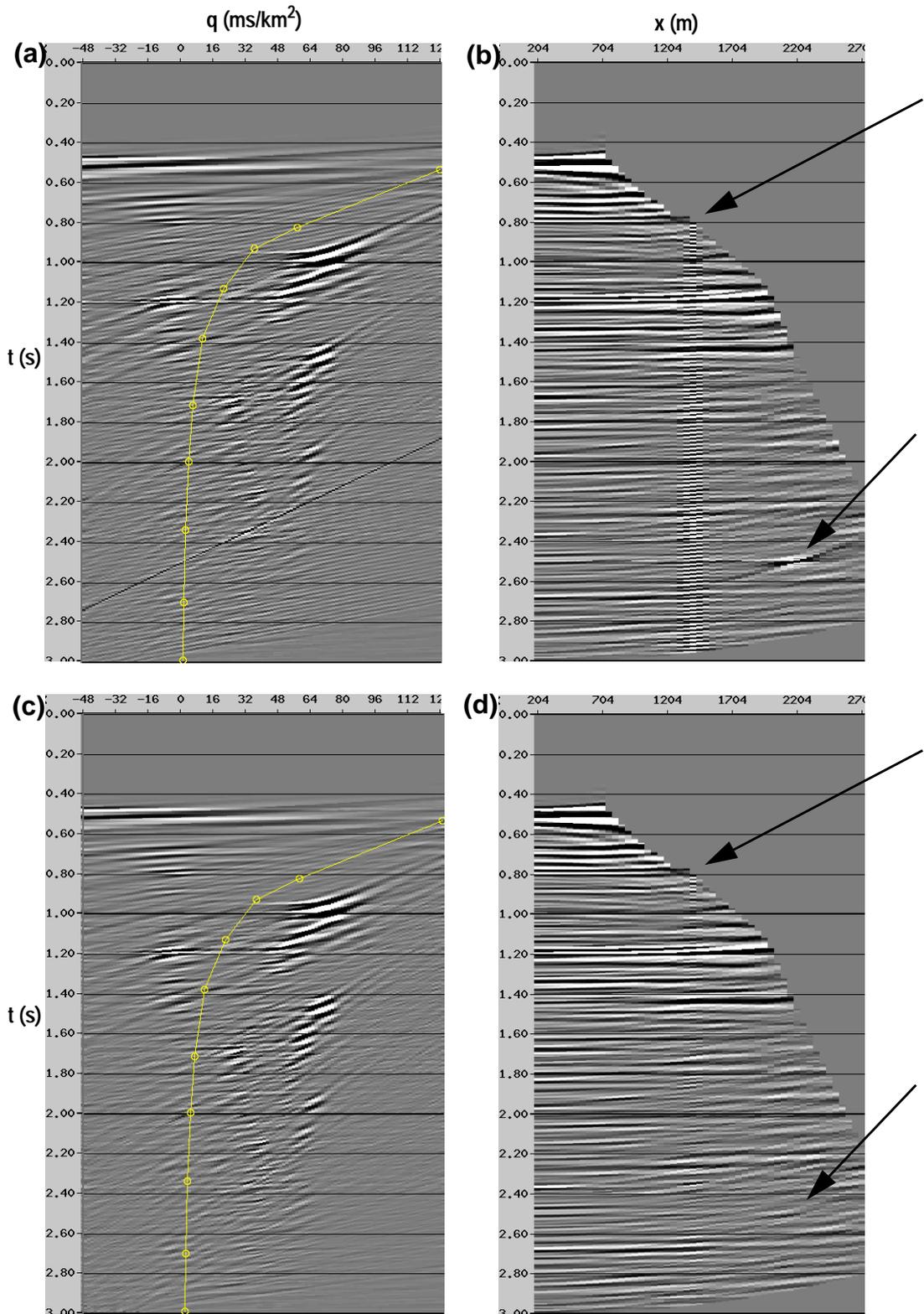


Figure 4.21 : PRD en norme L2 (a) et en norme L1 sur les données (c) de la collection CMP fig. 4.20. (b) (d) reconstruction du domaine des primaires (gauche du pointé) de chaque PRD.

4.3.3 - (Re-)conditionnement du problème

Le gradient correspondant au problème (E4.17) et (E4.19) à poids constant est:

$$-g(u) = \underline{L}^* \underline{P}_{Lp} (d - \underline{L}u) - \varepsilon \underline{I} \tag{E4.22}$$

Si on considère des évènements d'amplitudes différentes, le poids appliqué sur les résidus avant l'application de \underline{L}^* rehausse les amplitudes des évènements les plus faibles, entraînant un mauvais conditionnement du problème, les évènements affectés d'un poids élevé apparaissant en premier dans le modèle. Les données de la figure 4.23a contiennent des évènements d'amplitudes très différentes, différences qui se retrouvent sur le poids déduit des données et utilisé pour la première décomposition (4.23b). Dans le gradient de la première itération (4.23d, mais cela est valable pour les itérations suivantes), l'application du poids a égalisé tous les évènements, entraînant une mauvaise convergence. Afin de compenser cet effet, on peut calculer pour chaque point du modèle le poids moyen appliqué le long de la trajectoire correspondante dans les données, et diviser le gradient par ce poids moyen. Appelons \underline{P}_C ce poids de préconditionnement (diagonal). Le gradient préconditionné est:

$$-g_p(u) = \underline{P}_C^{-1} (\underline{L}^* \underline{P}_{Lp} (d - \underline{L}u) - \varepsilon \underline{I}) \tag{E4.23}$$

En figure 4.23c se trouve le poids de préconditionnement déduit du poids de la norme Lp, et en 4.23e le gradient préconditionné, dans lequel les amplitudes des évènements ont été rééquilibrés.

Ce préconditionnement améliore sensiblement la convergence (fig. 4.24). Cette amélioration devrait normalement être d'autant plus importante que les données présentent des évènements d'amplitudes très différentes.

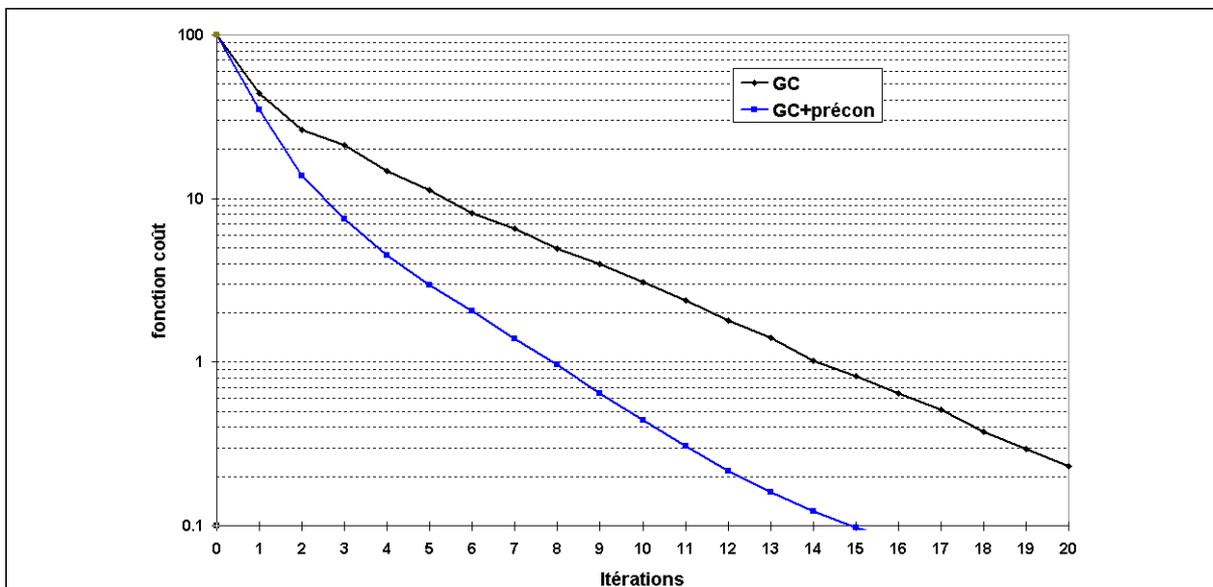
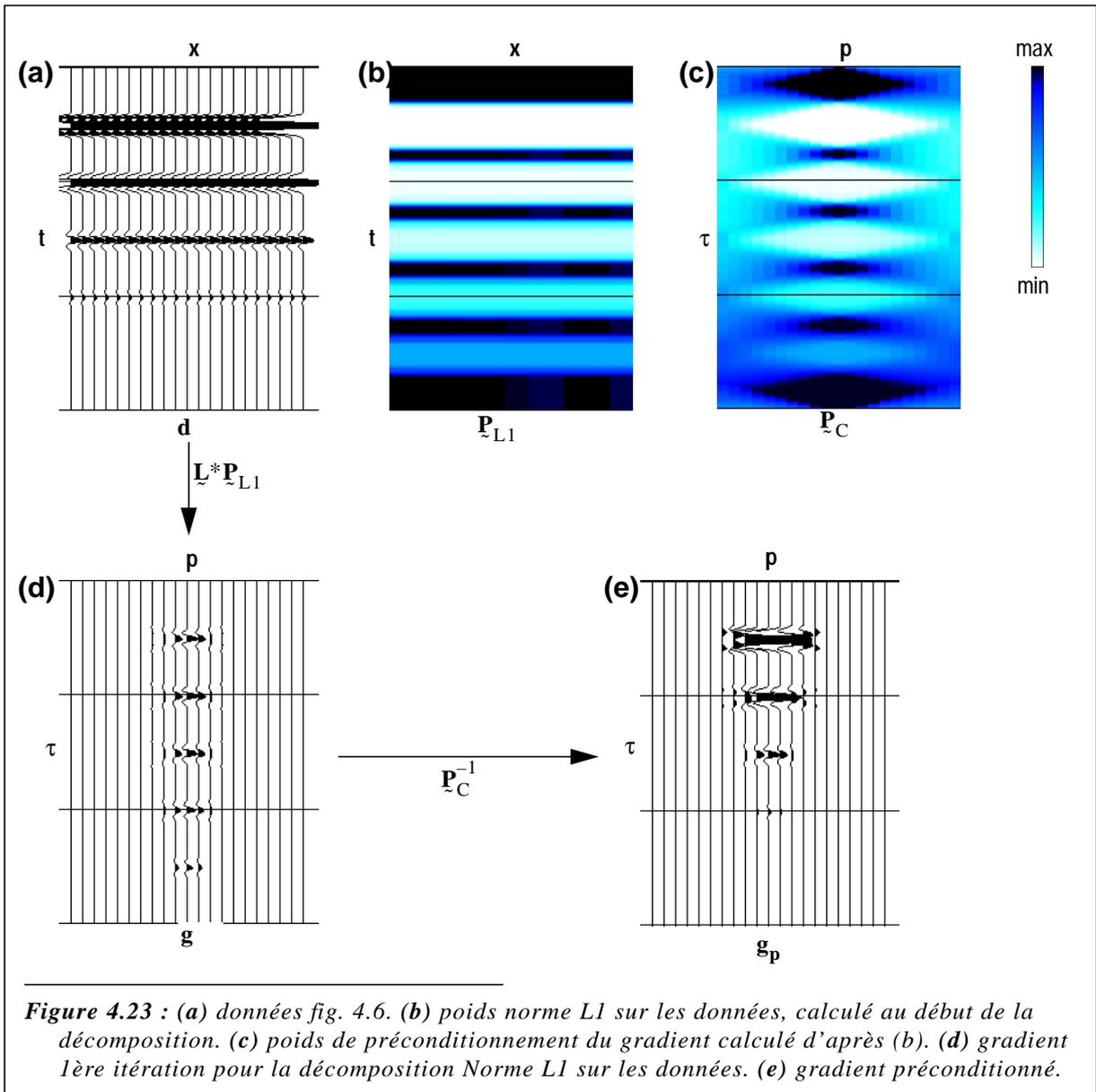


Figure 4.24 : PRD en norme L1 sur les données de la fig. 4.21c: convergence du gradient conjugué de la première décomposition avec et sans poids de préconditionnement pour compenser le poids sur les données.



4.4 - Poids sur le modèle calculé d'après les rapports Signal/Bruit

4.4.1 - Introduction

Dans sa thèse, Thorson (1984) a présenté une inversion stochastique pour la décomposition de Radon. La fonction à minimiser est (tirée de (E3.2) avec un modèle à priori est choisi nul):

$$\varphi(\mathbf{u}) = \frac{1}{2}(\mathbf{d} - \underline{\mathbf{L}}\mathbf{u}, \underline{\mathbf{C}}_d^{-1}(\mathbf{d} - \underline{\mathbf{L}}\mathbf{u})) + \frac{1}{2}(\mathbf{u}, \underline{\mathbf{C}}_u^{-1}\mathbf{u}) \quad (\text{E4.24})$$

Il s'agissait d'estimer la matrice de covariance sur le modèle d'après le modèle lui-même: le modèle à priori est toujours choisi nul, et la matrice de covariance diagonale, les variances de chaque échantillon du modèle étant estimées égales au carré de cet échantillon, ou plutôt en prenant une moyenne glissante

en temps sur la longueur T de l'ondelette sismique (lissage du poids pour tenir compte de la dépendance entre échantillons-temps proches):

$$\boxed{(\mathbf{C}_{\mathbf{u}})_{ii} = \tilde{\sigma}_{\mathbf{u}i}^2 = \bar{\mathbf{u}}_i^2 \quad \bar{\mathbf{u}}_i^2 = \frac{1}{T} \int_{t_i-T}^{t_i+T} \mathbf{u}(t_0, x_0)^2 dt_0} \quad (\text{E4.25})$$

Cette implémentation revient plus ou moins à appliquer une norme Lp (p=0) sur le modèle, tout en notant l'abus de langage, la norme Lp définie en (E4.3) n'étant en fait une norme que pour p>=1.

La matrice de covariance sur les données est choisie diagonale, à diagonale constante (le bruit sur les données est donc supposé gaussien, non corrélé, de variance uniforme σ_n^2 à déterminer):

$$\boxed{\tilde{\mathbf{C}}_{\mathbf{d}} = \sigma_n^2 \mathbf{Id}} \quad (\text{E4.26})$$

Les poids à appliquer pour résoudre le problème par un algorithme IRLS se déduisent des matrices de covariances. On introduit comme d'habitude une petite valeur pour borner le poids:

$$(\tilde{\mathbf{P}}_{\mathbf{u}})_{ii} = p(\bar{\mathbf{u}}_i) = (\mathbf{C}_{\mathbf{u}}^{-1})_{ii} = \frac{1}{\tilde{\sigma}_{\mathbf{u}i}^2} = \frac{1}{(\bar{\mathbf{u}}_i + \eta)^2} \quad (\text{E4.27})$$

Ce poids, même majoré, ne satisfait pas aux conditions suffisantes (E3.35) pour assurer la convergence des algorithmes IRLS. En particulier $\bar{\mathbf{u}}_i(\tilde{\mathbf{P}}_{\mathbf{u}})_{ii}$ n'est pas croissante par rapport à $\bar{\mathbf{u}}_i$, ce qui traduit le fait que la fonction à minimiser n'est pas convexe. La fonction que Thorson minimise est fonction du logarithme des échantillons du modèle:

$$\begin{aligned} \varphi_{\text{th.}}(\mathbf{u}) &= \frac{1}{2\sigma_n^2} \|\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}\|^2 + \sum_{i=1}^n f(\bar{\mathbf{u}}_i) \\ f(u) &= \int u p(u) du = \ln\left(\frac{|u|}{\eta} + 1\right) - \frac{|u|}{|u| + \eta} \end{aligned} \quad (\text{E4.28})$$

Ce qui correspond approximativement à minimiser l'entropie (ce serait 'exactement' l'entropie pour $\eta = 0$) du modèle lissé. Il faut prendre ici l'entropie du modèle au sens de la somme des entropies de chacun des échantillons pris comme une variable aléatoire gaussienne (l'entropie d'une variable aléatoire à moyenne nulle et de variance σ^2 étant égale à $\ln(2\pi e \sigma)$ (Burg, 1975)).

Une fois de plus, le fait de ne pas remplir les conditions (E3.35) ne signifie pas nécessairement que la convergence n'est pas possible, ces conditions n'étant que suffisantes, pas nécessaires.

La convergence pour un tel problème est très mauvaise, la dynamique du poids, donc le nombre de conditionnement du système, étant très élevée. D'une manière plus générale, les défauts (sensibilité au niveau de bruit estimé, atténuation des événements de faibles amplitudes) et les avantages (focalisation, anti-aliasing spatial) des normes Lp (p<2) se retrouvent mais extrêmement amplifiés. Néanmoins, utilisée sur des données adaptées cette inversion donne des résultats séduisants (fig. 4.25). Le résultat montré a tout de même nécessité 100 itérations de gradient conjugué, sans arriver pour autant au point de convergence (les résidus sont stables à partir de la 20ème itérations, mais pas le modèle).

4.4.2 - Estimation des variances par calcul des rapports Signal/Bruit

Dans (E4.27), l'estimation des variances des échantillons du modèle se fait à partir du modèle lui-même pendant les itérations. Cette modification du poids pendant les itérations est en grande partie

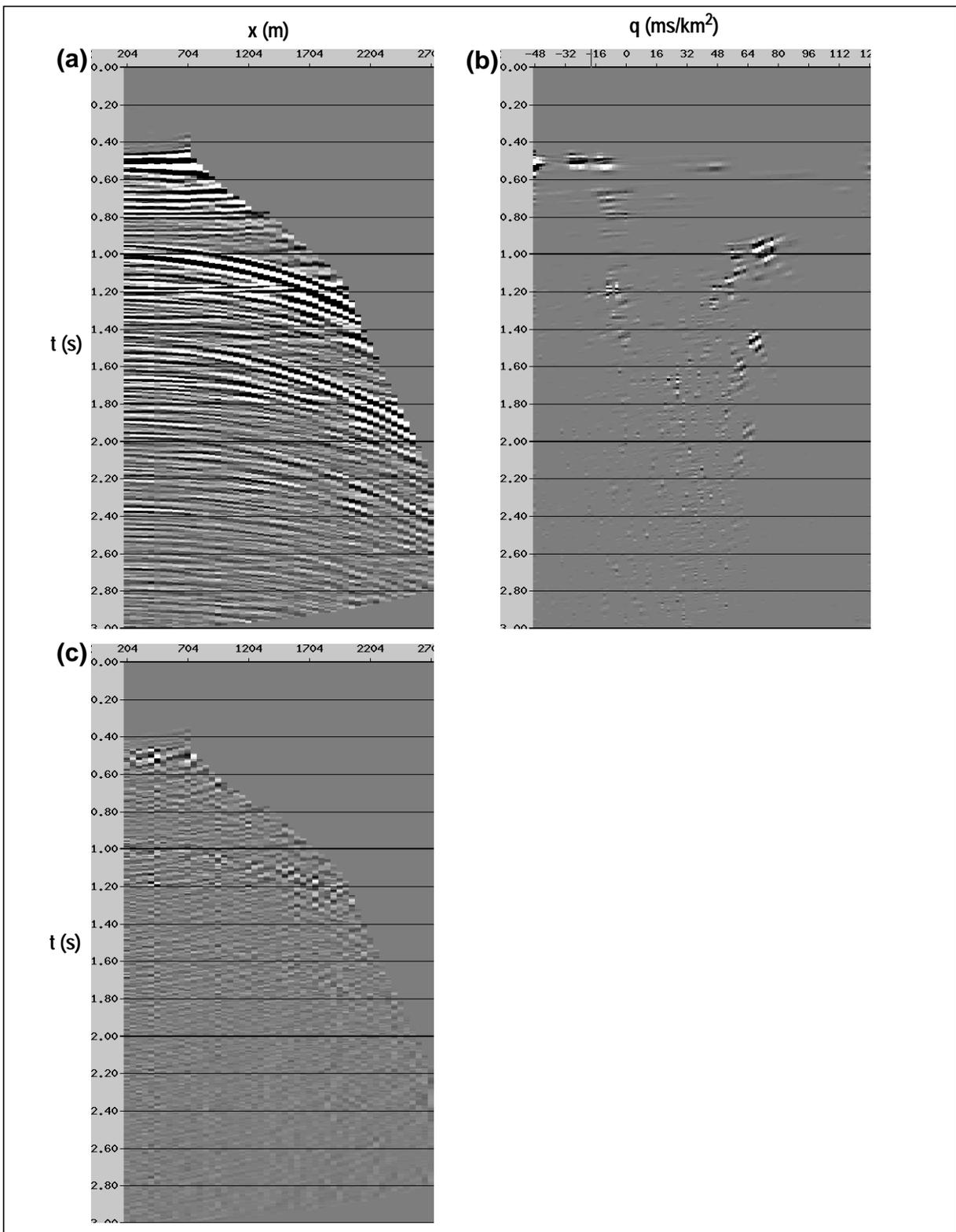


Figure 4.25 : (a) données figure 4.14; (b) PRD de (a) par inversion stochastique; (c) résidus

responsable de la lenteur de convergence. On peut garder l'esprit de l'inversion stochastique -qui consiste à estimer les variances dans le modèle-, mais en évitant de modifier cette estimation pendant les itérations. On peut les déduire par exemple d'un modèle obtenu par inversion en norme L2 (qui ne réclame que quelques itérations, surtout si ne on cherche pas une solution exacte), ou en analysant les amplitudes sur les données elles-mêmes, le long des trajectoires correspondant aux échantillons du modèle. Cette analyse le long des trajectoires peut également permettre de calculer une estimation du bruit sur les données, estimation faite pour chaque échantillon du modèle.

La fonction à minimiser est:

$$\phi_{\mathbf{P}_u}(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \mathbf{L}\mathbf{u}\|^2 + \frac{1}{2} (\mathbf{u}, \mathbf{P}_u \mathbf{u}) \quad (\text{E4.29})$$

où les éléments diagonaux du poids sont les rapports entre les variances estimées du bruit et les variances estimées dans le modèle:

$$\boxed{\mathbf{P}_u = \frac{\tilde{\sigma}_{ni}^2}{\tilde{\sigma}_{ui}^2}} \quad (\text{E4.30})$$

A chaque élément \mathbf{u}_i du modèle correspond une trajectoire dans les données (une droite pour la LRD, une parabole pour la PRD,...) qui coupe N_i traces valides, donc qui fait rentrer en jeu N_i échantillons $\text{traj}(\mathbf{d})_{ij}$, ($j=1, \dots, N_i$) (ces échantillons ne correspondent en général pas à une position temps échantillonnée exacte, mais sont interpolés). Le long de chaque trajectoire i on peut estimer une moyenne et une variance sur ces échantillons:

$$\begin{aligned} \mu(\text{traj}(\mathbf{d})_i) &= \frac{1}{N_i} \sum_{j=1}^{N_i} \text{traj}(\mathbf{d})_{ij} \\ \sigma^2(\text{traj}(\mathbf{d})_i) &= \frac{1}{N_i} \sum_{j=1}^{N_i} \text{traj}(\mathbf{d})_{ij}^2 - \mu^2(\text{traj}(\mathbf{d})_i) \end{aligned} \quad (\text{E4.31})$$

L'estimation de la variance du bruit dans (E4.30) peut être prise égale à la variance calculée dans (E4.31), après application d'un lissage (moyenne glissante) sur la longueur T de l'ondelette sismique:

$$\boxed{\tilde{\sigma}_{ni}^2 = \overline{\sigma^2(\text{traj}(\mathbf{d})_i)}} \quad (\text{E4.32})$$

4.4.2.1 - Variances du modèle estimées par la solution en norme L2

Les variances pour les échantillons du modèle peuvent être, comme nous l'avons envisagé ci-dessus, estimées d'après la solution (approximative, donc calculée avec peu d'itérations) en norme L2 (lissée également):

$$\boxed{\begin{aligned} \tilde{\sigma}_{ui}^2 &= (\mathbf{u}_{L2})_i^2 \\ \mathbf{u}_{L2} &\approx (\mathbf{L}^* \mathbf{L} + \varepsilon \mathbf{I})^{-1} \mathbf{L}^* \mathbf{d} \end{aligned}} \quad (\text{E4.33})$$

Afin que le poids calculé dans (E4.30) soit minoré, on introduit une minoration de l'estimation de la variance bruit par rapport au signal:

$$\boxed{\tilde{\sigma}_{ni}^2 = \max\left(\overline{\sigma^2(\text{traj}(\mathbf{d})_i)}, \left(\frac{\overline{\mu(\text{traj}(\mathbf{d})_i)}}{\lambda}\right)^2\right)} \quad (\text{E4.34})$$

avec pour valeur typique $\lambda = 10$.

Cette formulation a deux intérêts majeurs. D'une part, le poids appliqué au niveau du pic central d'un évènement ne dépend pas (comme c'est le cas en appliquant la norme L1 par exemple) de l'amplitude

absolue de l'évènement, mais du rapport S/B le long de celui-ci dans les données, rendant inutile une égalisation en temps des données ou du poids (avec la norme Lp, ce sont les évènements de forte amplitude qui sont favorisés dans le gradient; ici ce sont les évènements à fort rapport S/B). D'autre part le problème de l'estimation du niveau de bruit est implicitement résolu: on n'a pas à ajuster à priori de facteur d'amortissement d'après une proportion de résidus estimée globalement à priori. Le niveau de bruit est déduit des données elles-mêmes, et ce localement pour chaque échantillon du modèle. On ainsi une adaptation 'naturelle' au niveau de bruit.

Dans une PRD calculée avec cette méthode (fig. 4.26 (série 1)), il reste dans les résidus beaucoup d'évènement cohérents. L'estimation des variances du bruit sur les données est en effet perturbée par les interférences entre les évènements, conduisant à une surestimation, donc à une surestimation du poids, donc à des résidus trop importants. Cela est particulièrement sensible pour un petit évènement interférent avec un gros: la présence de ce dernier induit des grosses variations d'amplitudes le long du petit évènement, conduisant à une variance estimée importante, même en l'absence de bruit.

On réestime donc les variances du bruit à partir des résidus obtenus, ce qui permet de calculer un nouveau poids avec lequel on refait quelques itérations. On peut boucler plusieurs fois ce processus. Sur l'exemple de la figure 4.26, on réestime deux fois le niveau de bruit, et à chaque fois on effectue une série de 5 itérations.

Le processus est donc le suivant:

- (i) calcul d'une solution approximative en norme L2
- (ii) estimation des variances dans le modèle à partir de cette solution (équ. (E4.33))
- (iii) estimation du niveau de bruit sur les résidus (donc sur les données pour le premier passage) (équ. (E4.34))
- (iv) calcul du poids (équ. (E4.30))
- (v) Itérations à poids constant
- (vi) aller en (iii) ou arrêter.

On remet sur la figure 4.27 la PRD calculée suivant ce principe en figure 4.26. Le nombre total d'itérations dans cet exemple est de 2 (solution norme L2) + 3x5 (3 séries de 5 itérations à poids constant) + 3 (l'estimation du niveau de bruit est équivalent à une itération en temps de calcul), soit au total 20 'itérations'.

4.4.2.2 - Variances du modèle estimée par un simple stack sur les résidus

Si on veut éviter de calculer la décomposition préalable en norme L2 pour avoir une estimation des variances dans le modèle, on peut utiliser le calcul de l'amplitude moyenne le long de chaque trajectoire (équ. (E4.31)). Ce calcul n'est en effet rien d'autre que le stack (opérateur \mathbf{L}^*) divisé par la couverture (nombre d'échantillons réellement utilisés). L'estimation des variances dans le modèle est donc:

$$\tilde{\sigma}_{\mathbf{u}i}^2 = \overline{(\alpha N_i \mu(\text{traj}(\mathbf{d})_i))^2} \quad (\text{E4.35})$$

le scalaire α étant obtenu par recherche linéaire le long de la direction définie par $(\mathbf{g}_i = N_i \mu(\text{traj}(\mathbf{d})_i))$ pour minimiser $\left(\frac{1}{2} \|\mathbf{d} - \mathbf{L}(\alpha \mathbf{g})\|^2\right)$, ceci afin de mettre à l'échelle le modèle calculé.

L'intérêt principal de cette méthode est de réduire le temps de calcul. En effet le calcul des moyennes le long des trajectoires est de toutes façons nécessaire pour le calcul des variances (équ. (E4.31)), le seul surcoût provenant d'une application de l'opérateur \mathbf{L} dans la recherche linéaire.

On applique la méthode aux mêmes données que précédemment (fig. 4.28). Là encore, la PRD ainsi calculée laisse des résidus important (fig. 4.28, série 1). Il est donc nécessaire de reprendre une nouvelle décomposition sur les résidus. On réestime à partir de ceux-ci une variance du bruit. L'estimation des

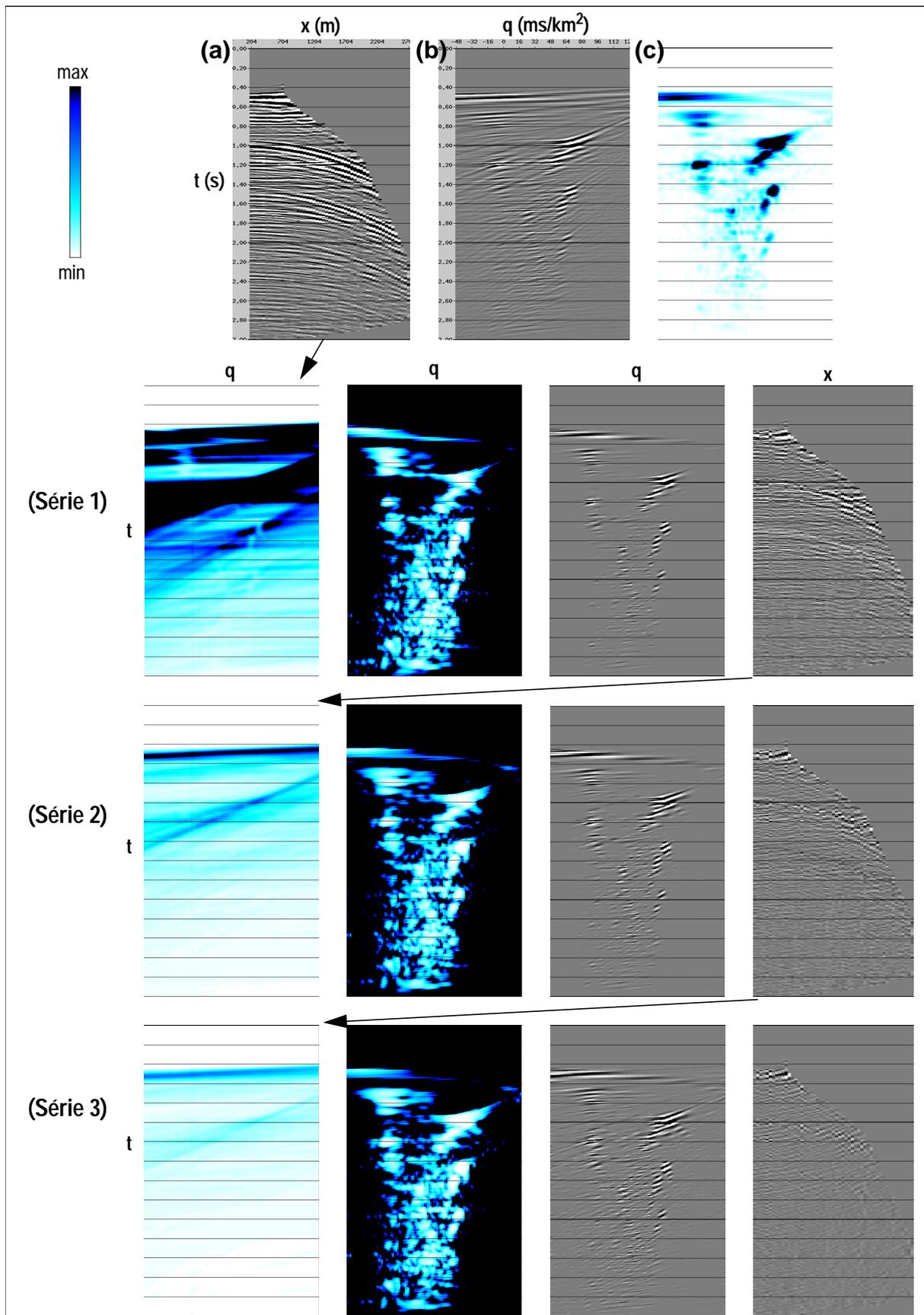


Figure 4.26 : (a) données dont on veut calculer la PRD. (b) PRD approximative en norme L2. (c) variances du modèle déduites de (b). **Séries 1,2,3:** (col.1) variances du bruit calculées sur les résidus (sur les données pour la série 1). (col.2) poids sur le modèle (rapport entre (col.1) et (c)). (col.3) PRD obtenue avec le poids de (col.2). (col.4) résidus. 5 Itérations dans chaque série à poids constant.

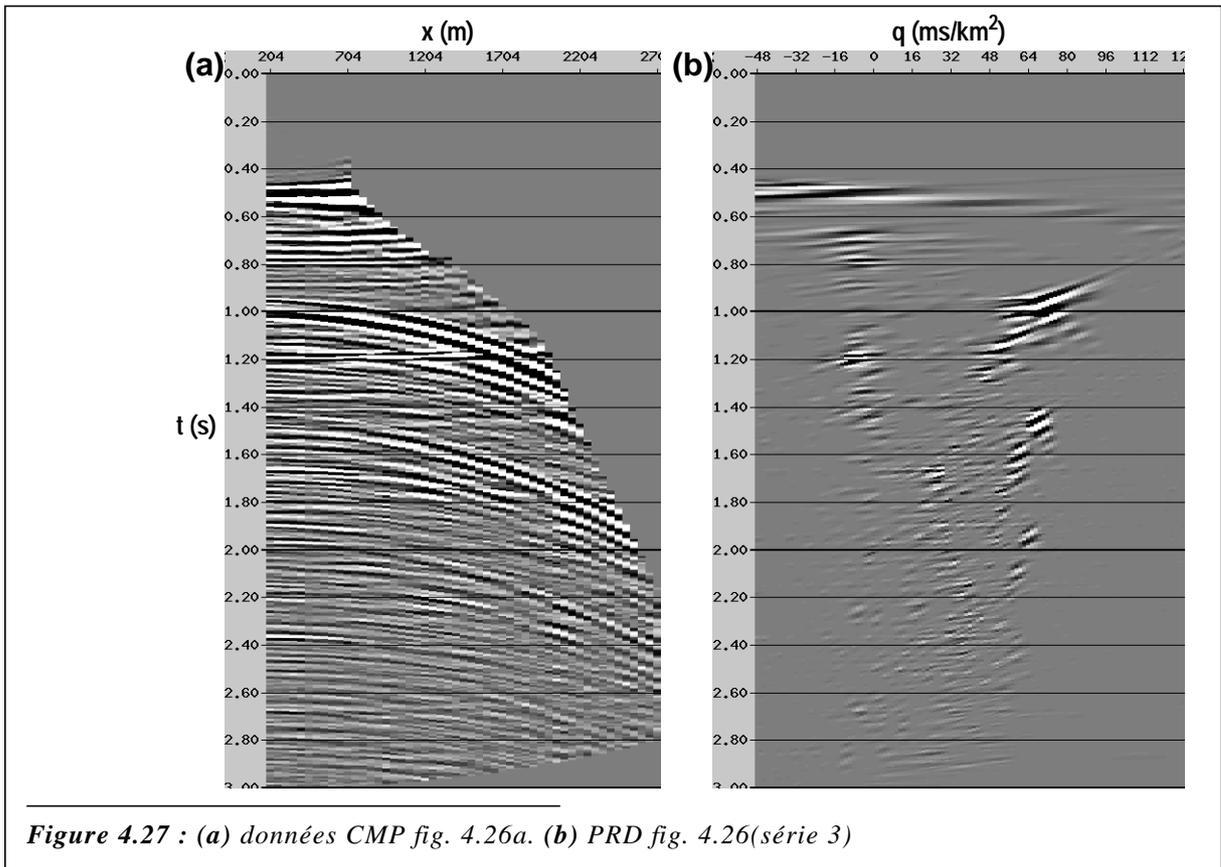


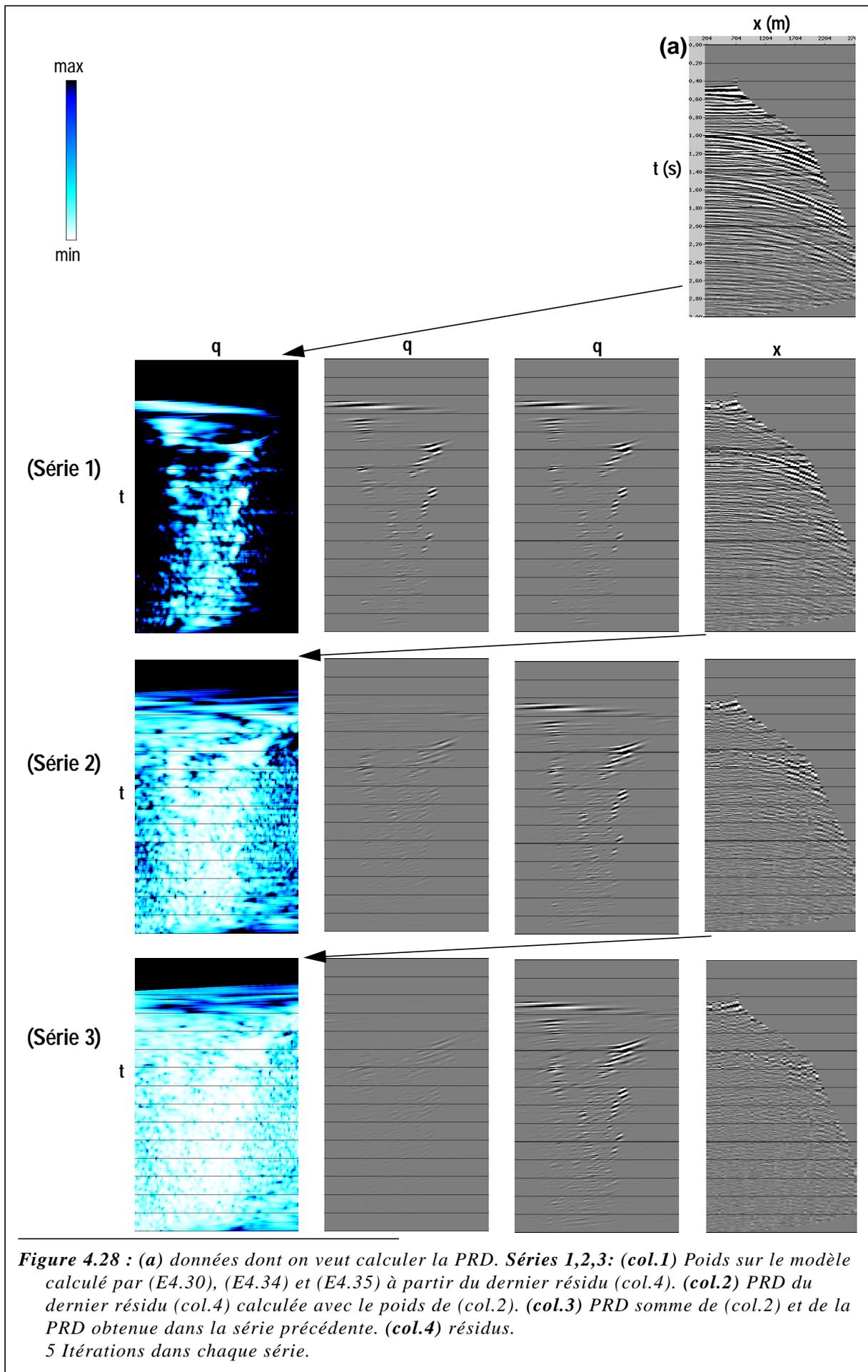
Figure 4.27 : (a) données CMP fig. 4.26a. (b) PRD fig. 4.26(série 3)

variances du modèle calculée par (E4.35) à partir des données se révèle trop peu précise pour pouvoir être réutilisée au cours de la nouvelle décomposition. On réestime donc également les variances du modèle à partir des résidus. Le poids qui en découle (équ. (E4.30)) ne correspond plus aux données de départ mais aux résidus, sur lesquels ont été calculées toutes les estimations. Il faut donc concevoir la nouvelle décomposition indépendamment de la précédente, les résidus constituant de nouvelles données à décomposer. Le résultat de cette décomposition s'ajoutant simplement à la décomposition précédente.

- (i) Initialisations: $\mathbf{u} \leftarrow \mathbf{0}$ $\mathbf{r}^{(0)} \leftarrow \mathbf{d}$ $k \leftarrow 0$
- (ii) $k \leftarrow k + 1$
- (iii) Calcul du poids sur le modèle (équ. (E4.31), (E4.34), (E4.35)) à partir de $\mathbf{r}^{(k-1)}$
- (iv) Décomposition de $\mathbf{r}^{(k-1)}$ avec ce poids: on obtient un modèle $\mathbf{u}^{(k)}$ et des résidus $\mathbf{r}^{(k)}$
- (v) $\mathbf{u} \leftarrow \mathbf{u} + \mathbf{u}^{(k)}$
- (vi) Aller en (ii) ou arrêter

La décomposition d'ordre (k) (étape (iv)) peut également être considérée comme la décomposition des données de départ \mathbf{d} , avec $\mathbf{u}^{(k-1)}$ comme modèle à priori (et comme point de départ de la décomposition) (Voir le paragraphe 3.4).

Sur la figure 4.28 se trouvent les décompositions successives et les résidus correspondants. Le résultat final se trouve sur la figure 4.27. Il diffère apparemment peu du résultat obtenu avec l'autre méthode (fig. 4.27). Le nombre total d'itérations dans cet exemple est de 3×5 (3 séries de 5 itérations à poids constant) + 4.5 (l'estimation du niveau de bruit et de la variance du modèle est équivalent à 1.5 itération en temps de calcul), soit au total 19.5 'itérations'.



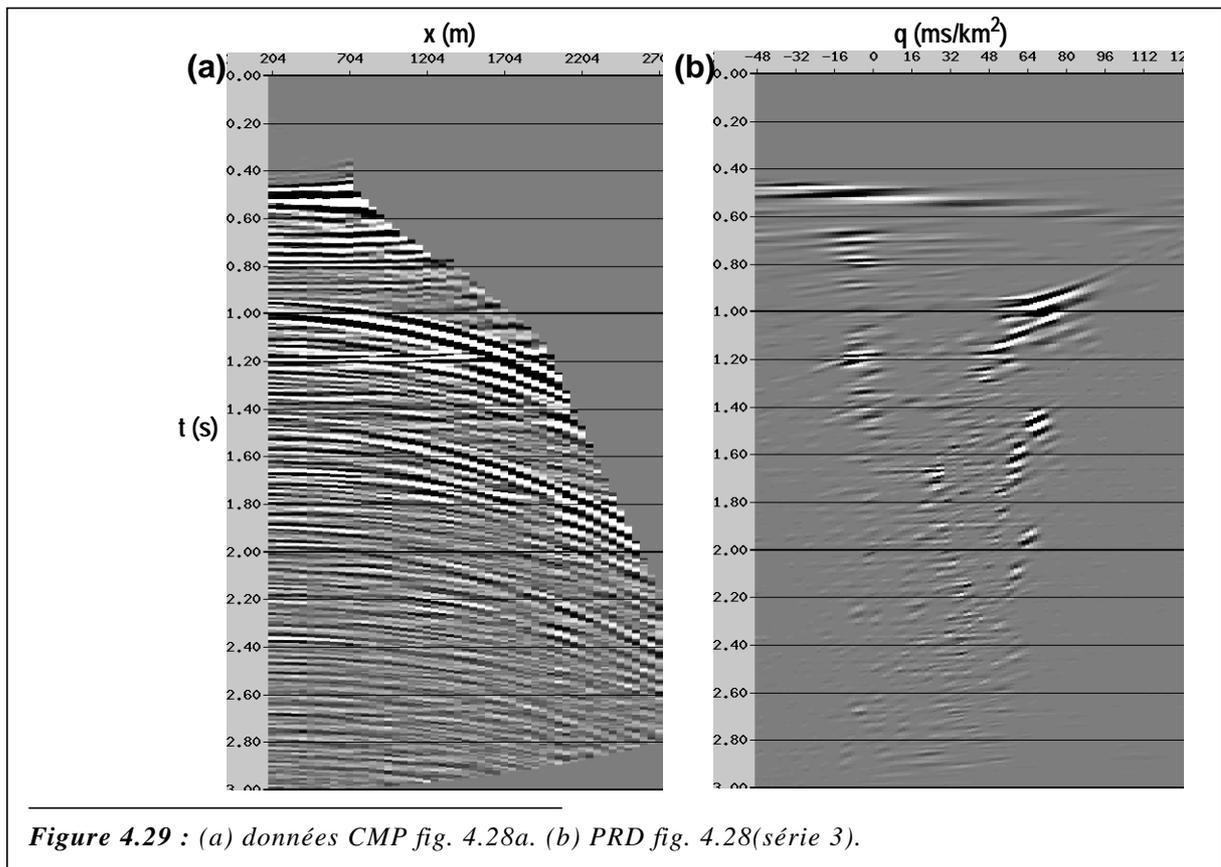


Figure 4.29 : (a) données CMP fig. 4.28a. (b) PRD fig. 4.28(série 3).

4.4.3 - Focalisation - Atténuation du bruit - perte de certaines données

La très grande focalisation obtenue par cette méthode est évidemment très intéressante pour l'application anti-multiples. On peut reprendre l'exemple de la figure 4.17 sur lequel on effectue une PRD suivie de la reconstruction de la partie du modèle contenant les événements horizontaux (fig. 4.30). Le calcul de la PRD par estimation des variances du bruit et du modèle (d'après la solution en norme L2) conduit à une excellente focalisation des événements (fig. 4.30a). La reconstruction des événements horizontaux est de la même qualité que celle que l'on obtient avec la décomposition en norme L1 (fig. 4.17b). Le principal avantage ici est de ne pas avoir à estimer à priori le niveau de bruit. A titre de comparaison, on a effectué également la PRD suivant le schéma décrit par Thorson (1985), dans lequel les variances du modèle ne sont pas fixées une fois pour toute mais déduites du modèle courant au cours des itérations. La focalisation est cette fois-ci totale, mais on note une atténuation énorme des petits événements noyés dans le bruit (fig. 4.30d-e). Sur la reconstruction le bruit a été atténué d'autant, de sorte que le rapport S/B sur ces événements est préservé (et même amélioré). Mais il peut être difficile d'accepter une telle distorsion des amplitudes relatives dans une même décomposition. Ce schéma est en fait extrême dans la parcimonie, avec les avantages et les inconvénients qui en découlent.

4.4.4 - Déplacement d'énergie dans le modèle

Les décompositions parcimonieuses n'étant pas linéaires (i.e. la décomposition n'est pas obtenue par l'application d'un opérateur linéaire), on constate parfois des effets indésirables. Ainsi au cours de l'optimisation, il peut être plus 'rentable' d'expliquer les données en déplaçant de l'énergie dans le modèle vers des positions à priori 'fausses', qui expliquent moins bien les données, mais moins pénalisantes parce que pondérées par un poids faible dans la fonction coût.

Reprenons par exemple les données de la figure 4.13a, en ajoutant un faible bruit gaussien (fig. 4.31a). Dans la PRD par estimation des rapports S/B (fig. 4.31b), on sépare comme à l'habitude les

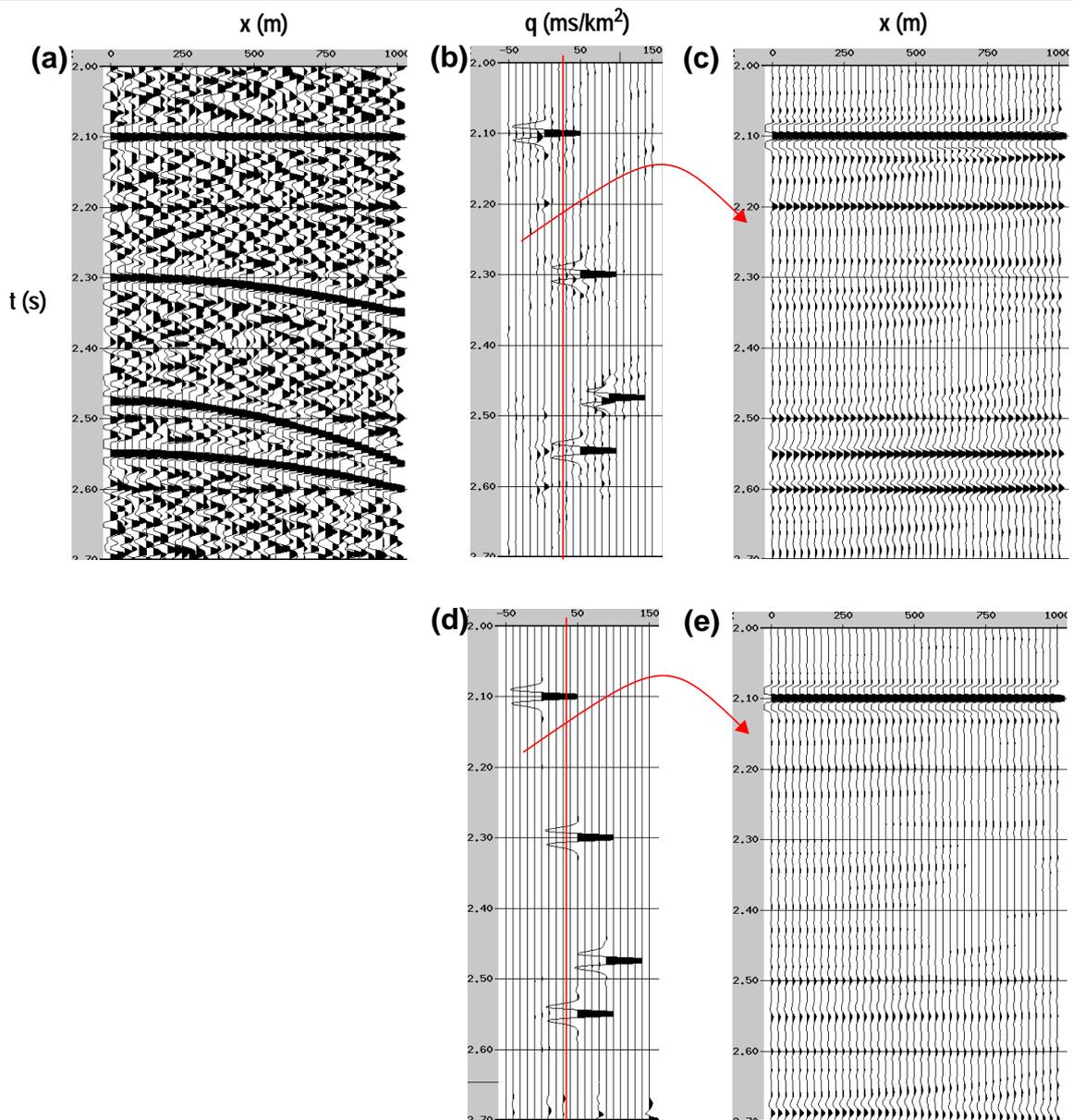
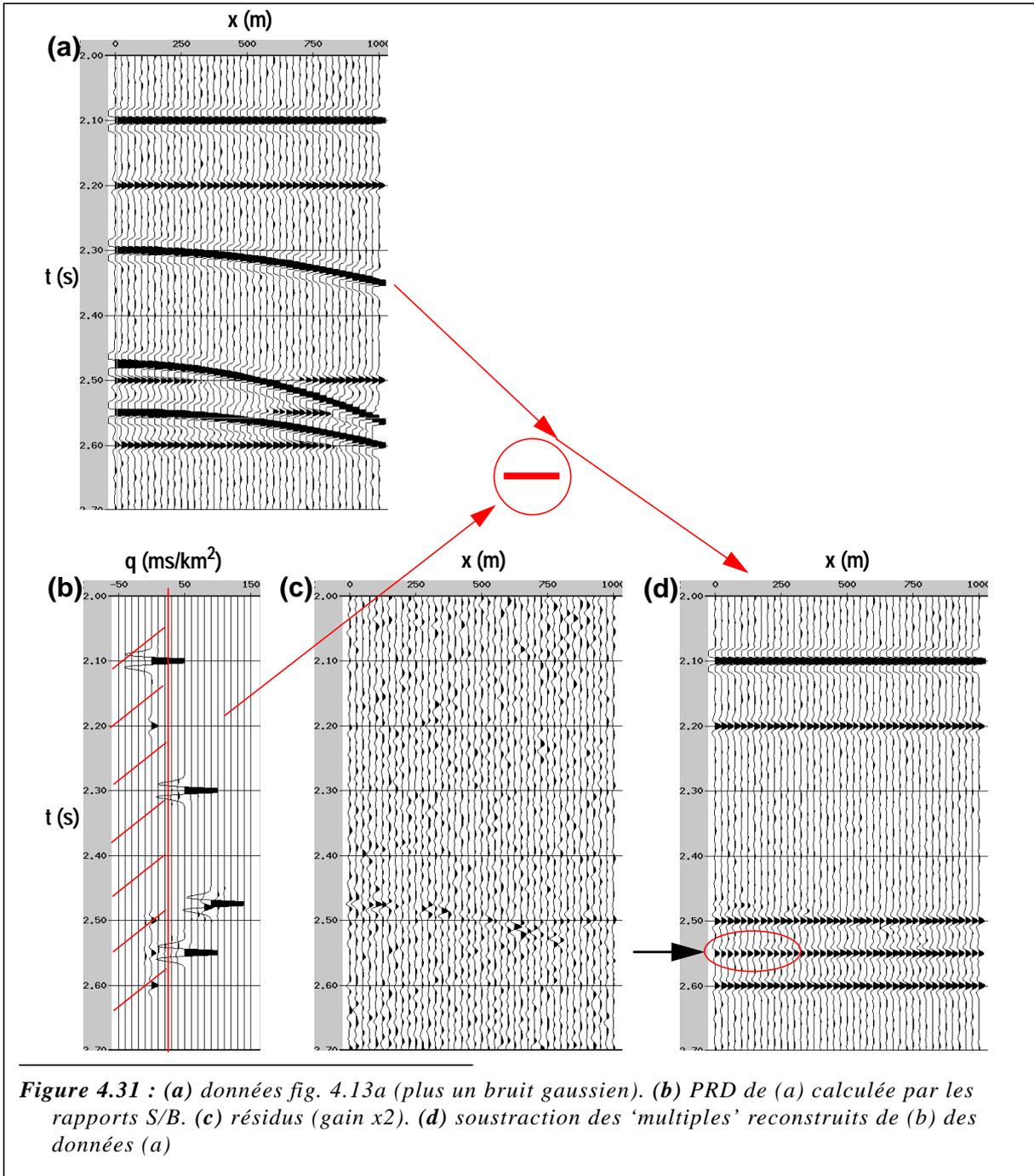


Figure 4.30 : (a): mêmes données que fig. 4.13a, entachées d'un bruit gaussien (de niveau comparable à l'amplitude des petits évènements). (b): PRD pondérée par les rapports S/B. (c): reconstructions du domaine de (b) $q < 25\text{ms/km}^2$. Gains=10 par rapport aux plus forts évènements. (d): PRD par inversion stochastique (Thorson). (e): reconstructions du domaine de (d) $q < 25\text{ms/km}^2$. Gains=10 par rapport aux plus forts évènements.

évènements horizontaux (les primaires) des évènements courbes (les multiples) par un mute. Au lieu de reconstruire directement les primaires, on reconstruit les multiples que l'on soustrait ensuite des données initiales (fig. 4.31d). Ce faisant, on ne risque en principe pas d'abîmer les amplitudes des primaires, dans le cas où ceux-ci auraient été mal modélisés (voir fig. 4.30 par exemple). Or l'amplitude de l'évènement horizontal à 2550ms est légèrement atténuée sur les faibles offsets. L'énergie manquante ne semble pas se trouver dans les résidus. La seule explication possible est que l'on a soustrait cette amplitude, donc que l'énergie correspondante se trouve dans la zone des multiples de la PRD ($q > 25\text{ms/km}^2$). Le primaire interfère en effet sur les faibles offsets avec un multiple d'amplitude 10 fois supérieure. La pondération (équ. (E4.30), (E4.33) et (E4.34)) appliquée dans le modèle sur le multiple ($q=50\text{ms/km}^2$) est donc beaucoup plus faible que sur le primaire ($q=0$). En conséquence une même amplitude sera beaucoup plus pénalisante (dans la fonction coût, équ. (E4.29)) placée dans le modèle à $q=0$ plutôt qu'à $q=50\text{ms/km}^2$. Or sur les faibles offsets, toutes les paraboles tendent à se confondre, ce qui signifie que

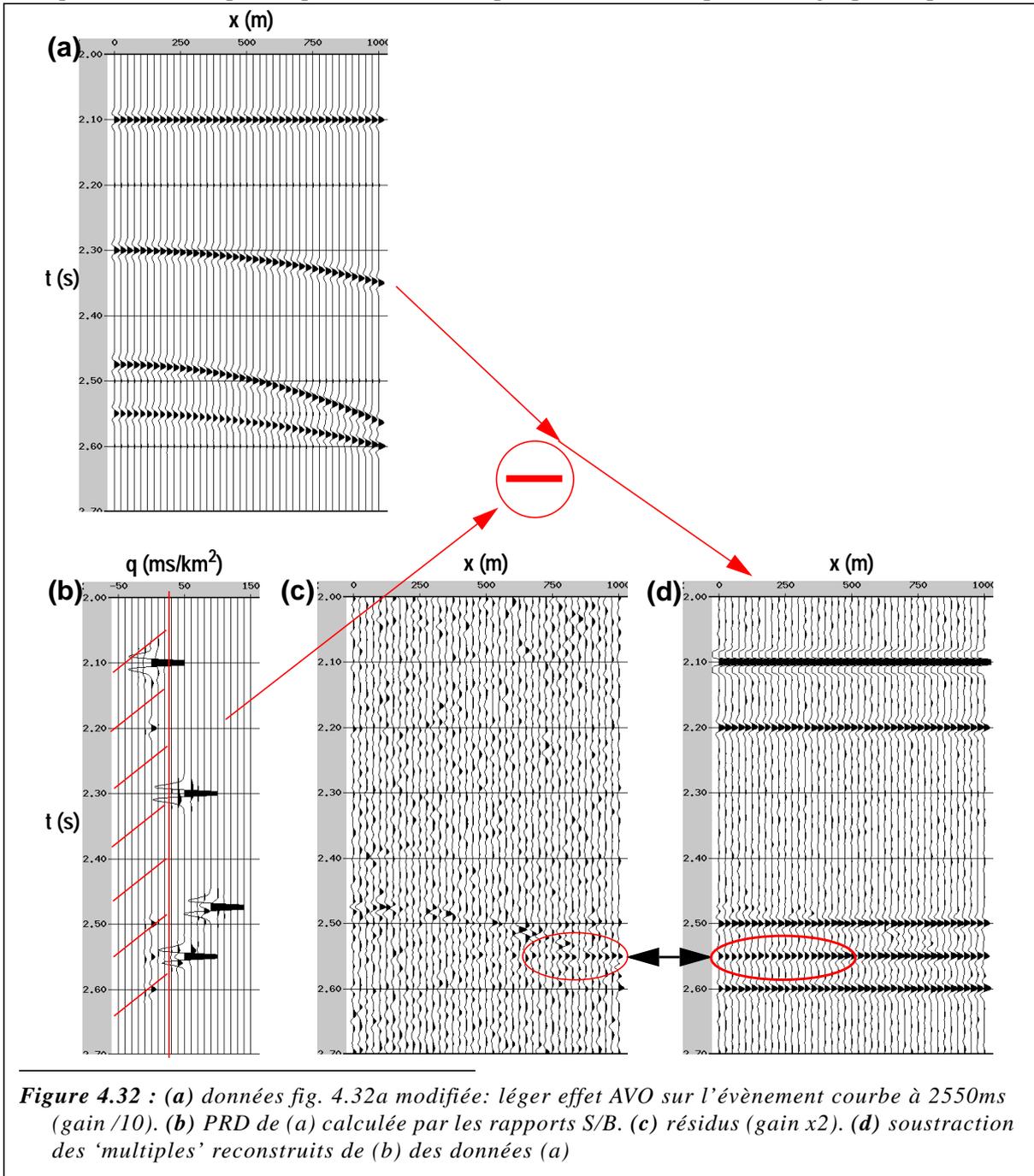
l'énergie dans le modèle qui explique l'énergie sur les faibles offsets peut être déplacée suivant q sans grandes conséquences. Dans le cas présent on assiste un déplacement de $q=0$ (position exacte) vers $q=50$ (position fausse).

Néanmoins, l'effet observé reste peu important: tous les événements sont parfaits (ou presque) du point de vue du modèle (paraboles d'amplitudes constantes), et le principe de parcimonie tend dans tous les cas à focaliser l'énergie dans le modèle, objectif plus difficile à atteindre en positionnant mal celle-ci.



Dégradons maintenant les données de ce point de vue en donnant un léger effet AVO à l'évènement courbe situé à 2550ms (fig. 4.32a, son amplitude n'est plus constante égale à 10.0, mais varie de 7.0 (offset min.) à 11.0 (offset max)) (en fonction de quoi sa représentation sur la PRD n'est d'ailleurs plus aussi compacte (fig. 4.32b)). L'atténuation d'amplitude sur les faibles offsets pour le primaire à 2550ms est cette fois-ci bien plus importante (fig. 4.32d). Explication: la décomposition parcimonieuse cherche à expliquer le multiple d'amplitude variable par le moins possible de paraboles d'amplitudes constantes. Pour cela, il 'affecte' une partie de l'énergie du primaire au multiple sur les faibles offsets (où ils sont

quasi-confondus) afin de réduire la variation d'amplitude le long de ce dernier. Ce faisant, le primaire n'est plus d'amplitude constante, puisqu'on lui a retiré une partie de son énergie sur les faibles offsets. L'optimisation 'résout le problème' en laissant dans les résidus une partie de son énergie sur les grands offsets (fig. 4.32c), afin encore une fois de ne représenter que des évènements d'amplitudes constantes. Tous ces effets se combinent pour arriver au meilleur compromis possible dans la fonction coût. Il est clair qu'un effet AVO plus important sur le multiple entraînerait un report d'énergie plus important.



Ce problème peut être plus ou moins gênant selon l'usage fait de la décomposition de Radon. Pour la séparation primaires/multiples il peut être catastrophique: à partir du moment où l'énergie d'un primaire a été déplacée dans le domaine des multiples, toute tentative d'atténuation des multiples affectera le primaire. Pour l'interpolation en revanche, cela ne devrait pas être trop grave: le déplacement d'énergie ne peut se faire que parce qu'il y a une certaine indétermination au départ. Dans l'exemple ci-dessus il s'agit de l'impossibilité de distinguer des paraboles proches sur les faibles offsets. Que l'énergie sur les offsets faibles soit alors affectée au primaire horizontal ou au multiple courbé ne change rien au résultat d'une interpolation à cet endroit.

4.5 - Conclusions

La minimisation de la norme L_p ($1 < p \leq 2$) du modèle se traduit dans la formulation des moindres carrés généralisés par l'application d'un poids sur le modèle dépendant du modèle lui-même. Ce poids entraîne une dégradation du conditionnement du problème.

Ce poids doit être lissé sur la durée du signal sismique afin de ne pas déformer ce dernier. Un algorithme IRLS avec plusieurs résolutions successives à poids constant est préférable à une mise à jour des poids pendant les itérations du gradient conjugué, car cela permet un meilleur contrôle de la convergence.

La focalisation des événements et la réduction de l'aliasing spatial obtenus avec cette décomposition découlent tous deux du principe de parcimonie, qui veut que l'on représente les données avec le moins possible d'échantillons significatifs du modèle.

Cette décomposition impose néanmoins une connaissance plus précise qu'avec la norme L_2 du niveau de bruit dans les données, à cause d'une plus grande sensibilité au réglage du facteur d'amortissement. Une trop grande valeur attribuée à ce dernier peut notamment conduire à la perte de certaines informations.

Cette connaissance du niveau de bruit n'est plus nécessaire si on estime le signal et le bruit par analyse des amplitudes le long de chaque courbe de la famille utilisée. Ces estimations permettent d'obtenir un poids à appliquer sur le modèle, poids adapté au niveau de bruit réel (les variations éventuelles de celui-ci dans les données sont implicitement prises en compte).

La minimisation de la norme L_p des résidus répond non pas à des considérations de parcimonie mais au besoin de traiter les bruits non gaussiens. Les techniques utilisées sont les mêmes que pour l'application de la norme L_p sur le modèle (algorithme IRLS, lissage du poids,...). Le seuillage imposé sur les faibles valeurs de résidus conduit à traiter en norme L_1 les résidus énergiques et en norme L_2 les résidus faibles. La dégradation du conditionnement du problème peut être compensée en partie par un préconditionnement adapté et déduit du poids appliqué sur les résidus.

Parmi les inconvénients des décompositions parcimonieuses, le plus gênant -car semble-t-il sans remède évident- est le déplacement d'énergie dans le modèle: l'énergie d'un événement peu énergétique peut être attribuée à un autre événement, proche et plus énergétique, parce que cette configuration est moins pénalisante dans la fonction coût. Les autres inconvénients sont relatifs (le temps de calcul supérieur doit être mis en balance avec l'amélioration de la qualité du résultat) ou ont des solutions (la sensibilité au niveau de bruit estimé peut être résolu par une estimation automatique des rapports S/B).

5 - Décompositions de Radon avec effet AVO

5.1 - Introduction

Les décompositions de Radon, telles que définies dans le chapitre 1 et utilisées dans ce rapport, utilisent des familles de courbes (ou de plans, pour le cas 3D, chapitre 6) d'amplitudes constantes. Elles permettent en général de décomposer des données contenant des événements qui ne correspondent pas forcément à ce modèle (cinématiques différentes, amplitudes variables), mais au prix d'une certaine complication de l'image obtenue. Pour les décompositions parcimonieuses, cela impose notamment une limite à la focalisation de l'énergie dans le modèle.

Or les événements présents dans les collections CMP présentent souvent des effets AVO (Amplitude Versus Offset), le coefficient de réflexion sur une interface géologique dépendant de l'angle d'incidence de l'onde sismique, donc de l'offset considéré. Des variations d'amplitudes avec l'offset peuvent également être introduites par certains traitements (une correction de divergence sphérique, pensée pour corriger correctement les primaires, conduit souvent à sur-corriger les multiples sur les offsets lointains).

Canadas and Dunand (1995), utilisent les décompositions paraboliques classiques des CMP pour extraire les informations d'AVO par une analyse de la forme des événements dans le modèle. Une autre approche consiste à utiliser directement des familles de courbes présentant des profils d'amplitudes correspondant aux événements à décomposer: on espère d'une part obtenir des décompositions plus propres et d'autre part extraire directement les informations d'AVO.

Nous utilisons ici en plus des courbes d'amplitudes constantes des courbes dont l'amplitude varie avec le carré de l'offset, permettant ainsi de modéliser des événements dont le profil d'amplitude peut s'écrire: $A(x) = v + w x^2$. v représente la partie constante; w n'est pas directement le coefficient AVO utilisé habituellement, mais lui est lié.

5.2 - Définitions

On reprend ici la plupart des définitions du paragraphe 1.3, en les simplifiant un peu:

Le support des données est un sous-ensemble de \mathbb{R}^2 :

$$D = \{(t, x); T_i \leq t \leq T_f; x_{\min} \leq x \leq x_{\max}\} \quad (E5.1)$$

De même pour le support des modèles:

$$U = \{(t_0, x_0); T_i \leq t_0 \leq T_f; x_{0\min} \leq x_0 \leq x_{0\max}\} \quad (E5.2)$$

L'espace des données ID est toujours l'ensemble des fonctions de carré intégrable sur D (équ. (E1.16)). Le produit scalaire et la norme choisis sur cet espace sont¹:

$$\begin{aligned} \mathbf{d}_1 \cdot \mathbf{d}_2 &= \frac{1}{T} \int_x \int_t \mathbf{d}_1(t, x) \mathbf{d}_2(t, x) dt dx \\ \|\mathbf{d}\|_{L_2} &= \sqrt{\mathbf{d} \cdot \mathbf{d}} \end{aligned} \quad (E5.3)$$

¹. La normalisation par $T = T_f - T_i$ a pour seul objet de simplifier certaines expressions par la suite.

L'espace des modèles IU est ici l'ensemble des fonctions à deux composantes $\mathbf{u}(t_0, x_0) = \begin{pmatrix} \mathbf{v}(t_0, x_0) \\ \mathbf{w}(t_0, x_0) \end{pmatrix}$, chacune étant de carré intégrable sur le support des modèles U. L'échantillon $\mathbf{u}(t_0, x_0)$ représente une courbe d'équation (l'application principale étant la PRD ou HRD des CMP. on utilise ici uniquement des familles de courbes invariantes par translation suivant le temps, équ. (E1.3)):

$$t = t_0 + h(x_0, x) \quad (\text{E5.4})$$

dont l'amplitude varie avec l'offset suivant la loi:

$$\boxed{A(t_0, x_0, x) = v(t_0, x_0) + w(t_0, x_0) x^2} \quad (\text{E5.5})$$

Le produit scalaire et la norme choisis sur cet espace sont:

$$\boxed{\begin{aligned} \mathbf{u}_1 \cdot \mathbf{u}_2 &= \frac{1}{T} \int_{x_0} \int_{t_0} (\mathbf{v}_1(t_0, x_0) \mathbf{v}_2(t_0, x_0) + \mathbf{w}_1(t_0, x_0) \mathbf{w}_2(t_0, x_0)) dt_0 dx_0 \\ \|\mathbf{u}\|_{L_2} &= \sqrt{\mathbf{u} \cdot \mathbf{u}} \end{aligned}} \quad (\text{E5.6})$$

La modélisation s'écrit toujours comme la superposition de toutes les courbes possibles:

$$(\underline{\mathbf{L}}\mathbf{u})(t, x) = \int_{x_0} \int_{t_0} (\mathbf{v}(t_0, x_0) + \mathbf{w}(t_0, x_0) x^2) \delta(t - (t_0 + h(x_0, x))) dt_0 dx_0 \quad (\text{E5.7})$$

L'intégration de la fonction de dirac aboutit à:

$$(\underline{\mathbf{L}}\mathbf{u})(t, x) = \int_{x_0} (\mathbf{v}(t - h(x_0, x), x_0) + \mathbf{w}(t - h(x_0, x), x_0) x^2) dx_0 \quad (\text{E5.8})$$

On peut séparer les deux composantes afin d'obtenir une écriture matricielle:

$$\boxed{\begin{aligned} \underline{\mathbf{L}}\mathbf{u} &= \begin{bmatrix} \underline{\mathbf{L}}_v & \underline{\mathbf{L}}_w \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{w} \end{bmatrix} \\ (\underline{\mathbf{L}}_v \mathbf{v})(t, x) &= \int_{x_0} \mathbf{v}(t - h(x_0, x), x_0) dx_0 \\ (\underline{\mathbf{L}}_w \mathbf{w})(t, x) &= \int_{x_0} \mathbf{w}(t - h(x_0, x), x_0) x^2 dx_0 \end{aligned}} \quad (\text{E5.9})$$

Par rapport aux produits scalaires utilisés, l'opérateur adjoint s'écrit:

$$\boxed{\begin{aligned} \underline{\mathbf{L}}^* \mathbf{d} &= \begin{bmatrix} \underline{\mathbf{L}}_v^* \\ \underline{\mathbf{L}}_w^* \end{bmatrix} \mathbf{d} \\ (\underline{\mathbf{L}}_v^* \mathbf{d})(t_0, x_0) &= \int_x \mathbf{d}(t_0 + h(x_0, x), x) dx \\ (\underline{\mathbf{L}}_w^* \mathbf{d})(t_0, x_0) &= \int_x \mathbf{d}(t_0 + h(x_0, x), x) x^2 dx \end{aligned}} \quad (\text{E5.10})$$

Chacune des composante s'obtient donc par sommation le long de la courbe considéré, avec une pondération dépendant de l'offset, égale en fait au profil d'amplitude représenté par cette composante.

5.3 - Conditionnement du problème par rapport aux deux composantes

On cherche ici à étudier les interactions entre les deux composantes du modèle. Supposons que le support des modèles est réduit à une seule valeur du paramètre x_0 . Toutes les équations sont les mêmes en supprimant simplement les intégrales par rapport à ce paramètre. En particulier:

$$\begin{aligned}
 (\underline{\mathbf{L}}_{\mathbf{v}}\mathbf{v})(t, x) &= \mathbf{v}(t - h(x)) \\
 (\underline{\mathbf{L}}_{\mathbf{w}}\mathbf{w})(t, x) &= \mathbf{w}(t - h(x))x^2 \\
 (\underline{\mathbf{L}}_{\mathbf{v}}^*\mathbf{d})(t_0) &= \int_x \mathbf{d}(t_0 + h(x), x)dx \\
 (\underline{\mathbf{L}}_{\mathbf{w}}^*\mathbf{d})(t_0) &= \int_x \mathbf{d}(t_0 + h(x), x)x^2 dx
 \end{aligned} \tag{E5.11}$$

L'expression de l'opérateur composé est dans ce cas très simple:

$$\begin{aligned}
 \underline{\mathbf{L}}^*\underline{\mathbf{L}} &= \begin{bmatrix} \underline{\mathbf{L}}_{\mathbf{v}}^*\underline{\mathbf{L}}_{\mathbf{v}} & \underline{\mathbf{L}}_{\mathbf{v}}^*\underline{\mathbf{L}}_{\mathbf{w}} \\ \underline{\mathbf{L}}_{\mathbf{w}}^*\underline{\mathbf{L}}_{\mathbf{v}} & \underline{\mathbf{L}}_{\mathbf{w}}^*\underline{\mathbf{L}}_{\mathbf{w}} \end{bmatrix} \\
 (\underline{\mathbf{L}}_{\mathbf{v}}^*\underline{\mathbf{L}}_{\mathbf{v}}\mathbf{v})(t_0) &= \int_x \mathbf{v}((t_0 + h(x)) - h(x))dx = \mathbf{v}(t_0) \int_x dx \\
 (\underline{\mathbf{L}}_{\mathbf{v}}^*\underline{\mathbf{L}}_{\mathbf{w}}\mathbf{w})(t_0) &= \int_x \mathbf{w}((t_0 + h(x)) - h(x))x^2 dx = \mathbf{w}(t_0) \int_x x^2 dx \\
 (\underline{\mathbf{L}}_{\mathbf{w}}^*\underline{\mathbf{L}}_{\mathbf{v}}\mathbf{v})(t_0) &= \int_x \mathbf{v}((t_0 + h(x)) - h(x))x^2 dx = \mathbf{v}(t_0) \int_x x^2 dx \\
 (\underline{\mathbf{L}}_{\mathbf{w}}^*\underline{\mathbf{L}}_{\mathbf{w}}\mathbf{w})(t_0) &= \int_x \mathbf{w}((t_0 + h(x)) - h(x))x^2 x^2 dx = \mathbf{w}(t_0) \int_x x^4 dx
 \end{aligned}$$

Posons les quantités suivantes, qui seront utiles par la suite:

$$\mathbf{X}_0 = \int_x dx = x_{\max} - x_{\min} \quad \mathbf{X}_2 = \int_x x^2 dx \quad \mathbf{X}_4 = \int_x x^4 dx \tag{E5.12}$$

L'opérateur composé s'écrit alors:

$$(\underline{\mathbf{L}}^*\underline{\mathbf{L}}\mathbf{u})(t_0) = \begin{bmatrix} \mathbf{X}_0 & \mathbf{X}_2 \\ \mathbf{X}_2 & \mathbf{X}_4 \end{bmatrix} \begin{pmatrix} \mathbf{v}(t_0) \\ \mathbf{w}(t_0) \end{pmatrix} \tag{E5.13}$$

Soit, avec un produit de Kronecker (voir nomenclature):

$$\tilde{\mathbf{L}}^* \tilde{\mathbf{L}} = \begin{bmatrix} \mathbf{X0Id} & \mathbf{X2Id} \\ \mathbf{X2Id} & \mathbf{X4Id} \end{bmatrix} = \mathbf{Id} \otimes \begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix} \quad (\text{E5.14})$$

Plus l'opérateur composé est proche de l'identité, meilleur est le conditionnement du problème. Sinon, on peut preconditionner le problème avec la meilleure approximation dont on dispose de l'inverse de l'opérateur composé. Or cet inverse est ici facile à calculer:

$$(\tilde{\mathbf{L}}^* \tilde{\mathbf{L}})^{-1} = \mathbf{Id} \otimes \begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix}^{-1} = \mathbf{Id} \otimes \left(\frac{1}{\mathbf{X0X4} - \mathbf{X2}^2} \begin{bmatrix} \mathbf{X4} & -\mathbf{X2} \\ -\mathbf{X2} & \mathbf{X0} \end{bmatrix} \right) \quad (\text{E5.15})$$

En effet:

$$\begin{aligned} \left(\mathbf{Id} \otimes \begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix}^{-1} \right) \left(\mathbf{Id} \otimes \begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix} \right) &= (\mathbf{IdId}) \otimes \left(\begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix} \right) \\ &= \mathbf{Id} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{Id} & \mathbf{0} \\ \mathbf{0} & \mathbf{Id} \end{bmatrix} \end{aligned}$$

Le poids de preconditionnement doit donc être:

$$\mathbf{P}_{\tilde{\mathbf{avo}}}^{-1} = \mathbf{Id} \otimes \left(\frac{1}{\mathbf{X0X4} - \mathbf{X2}^2} \begin{bmatrix} \mathbf{X4} & -\mathbf{X2} \\ -\mathbf{X2} & \mathbf{X0} \end{bmatrix} \right) \quad (\text{E5.16})$$

lemme: Ce poids est symétrique, défini, et positif.

Démonstration:

Raisonnons sur son inverse:
le déterminant est égal à:

$$\begin{aligned} \det \left(\begin{bmatrix} \mathbf{X0} & \mathbf{X2} \\ \mathbf{X2} & \mathbf{X4} \end{bmatrix} \right) &= \mathbf{X0X4} - \mathbf{X2}^2 \\ &= (x_{\max} - x_{\min}) \frac{(x_{\max}^5 - x_{\min}^5)}{5} - \left(\frac{x_{\max}^3 - x_{\min}^3}{3} \right)^2 \\ &= \frac{(x_{\max} - x_{\min})^4 (4x_{\max}^2 + 7x_{\min}x_{\max} + 4x_{\min}^2)}{45} \end{aligned}$$

plaçons nous dans le cas où $x_{\max} > x_{\min}$. Le premier terme est donc strictement positif. Considérons le second comme un polynôme du second degré en x_{\max} . Le discriminant de ce polynôme est: $\Delta = 49x_{\min}^2 - 64x_{\min}^2 < 0$. Le polynôme n'a donc aucune racine réelle (sauf pour le cas $x_{\min}=0$ en $x_{\max}=0$, qui n'est bien sûr pas retenu), et est donc toujours strictement positif (le coefficient dominant étant positif). Le déterminant est donc toujours strictement positif, ce qui prouve le lemme.

On peut noter que l'opérateur $\mathbf{P}_{\tilde{\mathbf{avo}}}^{-1} \tilde{\mathbf{L}}^*$ serait l'adjoint naturel de $\tilde{\mathbf{L}}$ si le produit scalaire utilisé était pondéré par le poids lui-même: $(\mathbf{u1}, \mathbf{u2}) = \mathbf{u1} \cdot \mathbf{P}_{\tilde{\mathbf{avo}}} \mathbf{u2}$:

$$\mathbf{d} \cdot \underline{\mathbf{L}}\mathbf{u} = \underline{\mathbf{L}}^* \mathbf{d} \cdot \mathbf{u} = \underline{\mathbf{L}}^* \mathbf{d} \cdot \left(\mathbf{P}_{\text{avo}}^{-1} \mathbf{P}_{\text{avo}} \right) \mathbf{u}$$

Le poids est symétrique, donc son inverse l'est également:

$$\mathbf{d} \cdot \underline{\mathbf{L}}\mathbf{u} = \left(\mathbf{P}_{\text{avo}}^{-1} \underline{\mathbf{L}}^* \mathbf{d} \right) \cdot \left(\mathbf{P}_{\text{avo}} \mathbf{u} \right)$$

D'où le résultat.

Or ce produit scalaire est logique à posteriori, car il traduit une équivalence de l'énergie entre les deux représentations (données et modèles):

Lemme:

$$\boxed{(\mathbf{u}_1, \mathbf{u}_2) = \mathbf{u}_1 \cdot \mathbf{P}_{\text{avo}} \mathbf{u}_2 = \underline{\mathbf{L}}\mathbf{u}_1 \cdot \underline{\mathbf{L}}\mathbf{u}_2} \quad (E5.17)$$

Démonstration:

$$\mathbf{u}_1 \cdot \mathbf{P}_{\text{avo}} \mathbf{u}_2 =$$

$$\int_{t_0} (X_0 \mathbf{v}_1(t) \mathbf{v}_2(t) + X_2 \mathbf{v}_1(t) \mathbf{w}_2(t) + X_2 \mathbf{w}_1(t) \mathbf{v}_2(t) + X_4 \mathbf{w}_1(t) \mathbf{w}_2(t)) dt$$

et d'autre part:

$$\begin{aligned} \underline{\mathbf{L}}\mathbf{u}_1 \cdot \underline{\mathbf{L}}\mathbf{u}_2 &= \iint_{t \ x} (\mathbf{v}_1(t) + \mathbf{w}_1(t)x^2)(\mathbf{v}_2(t) + \mathbf{w}_2(t)x^2) dx dt \\ &= \int_t \left(\mathbf{v}_1(t) \mathbf{v}_2(t) \int_x dx + (\mathbf{v}_1(t) \mathbf{w}_2(t) + \mathbf{w}_1(t) \mathbf{v}_2(t)) \int_x x^2 dx + \mathbf{w}_1(t) \mathbf{w}_2(t) \int_x x^4 dx \right) dt \\ &= \int_t (\mathbf{v}_1(t) \mathbf{v}_2(t) X_0 + (\mathbf{v}_1(t) \mathbf{w}_2(t) + \mathbf{w}_1(t) \mathbf{v}_2(t)) X_2 + \mathbf{w}_1(t) \mathbf{w}_2(t) X_4) dt \end{aligned}$$

Les deux expressions sont bien égales

On va donc résoudre le problème inverse en minimisant:

$$\boxed{\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \underline{\mathbf{L}}\mathbf{u}_1\|^2 + \frac{\varepsilon}{2} \mathbf{u} \cdot \mathbf{P}_{\text{avo}} \mathbf{u}} \quad (E5.18)$$

On étend bien sûr l'utilisation de ce poids dans le cas général où le paramètre x_0 n'est pas réduit à une seule valeur.

5.4 - Décomposition en norme L2

On utilise des données synthétiques comportant 4 événements horizontaux, de profils d'amplitudes variés (fig. 5.1):

- 200ms: $A_1(x) = 1.0 + 0.0 (x/1000)^2$
- 400ms: $A_2(x) = 0.0 + 2.0 (x/1000)^2$
- 600ms: $A_3(x) = 2.0 - 2.0 (x/1000)^2$
- 800ms: $A_4(x) = 2.0 - 4.0 (x/1000)^2$

Les PRD en norme L2 se trouvent sur la figure 5.2: on peut comparer la PRD classique (paraboles d'amplitudes constantes) et la PRD avec effets AVO (PRDavo).

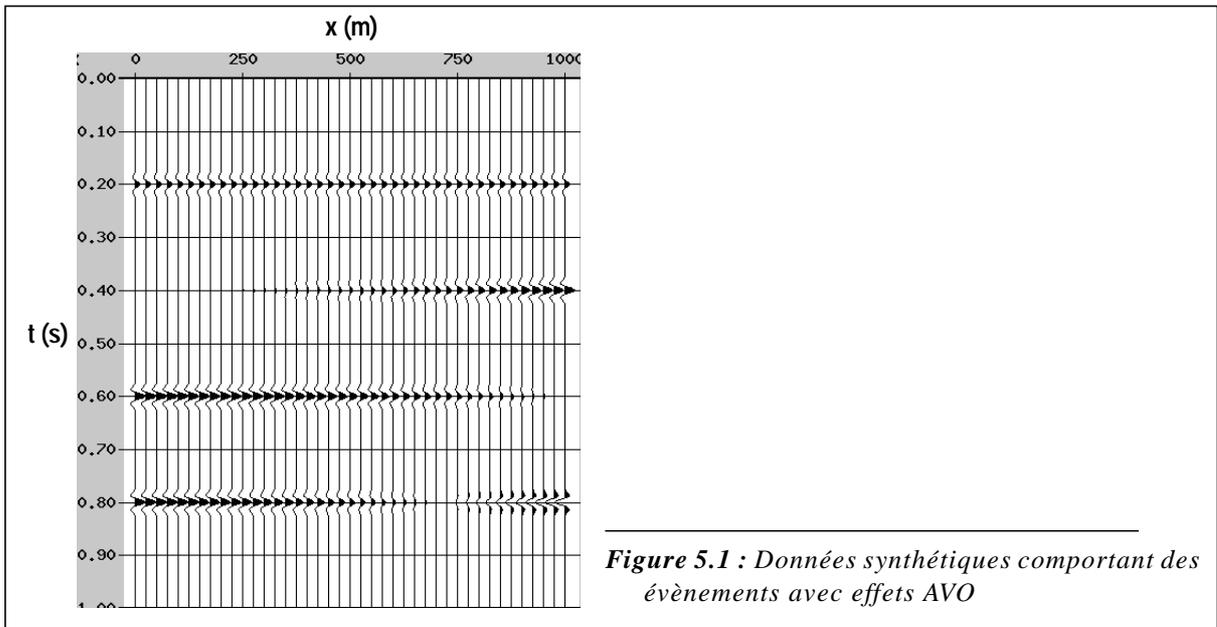


Figure 5.1 : Données synthétiques comportant des évènements avec effets AVO

On utilise les deux familles de paraboles décrites dans le paragraphe 5.2. Le coefficient est normalisé par rapport à l’offset maximum sur la courbe considérée. L’amplitude s’écrit donc:

$$A(x) = v + w \left(\frac{x}{x_{\max}} \right)^2 .$$

Les deux décompositions laissent peu de résidus. Dans la PRD classique, les formes des évènements dans le modèle sont très affectées pour pouvoir bien représenter les variations d’amplitudes. Pour l’évènement à t=800ms, la trace q=0 ne porte quasiment aucune énergie, et il est donc impossible de le pointer correctement.

Sur la PRDavo le résultat n’est pas tout à fait à la hauteur des espérances. Pour le premier évènement (d’amplitude constante), on voit apparaître beaucoup d’énergie dans la composante d’amplitude variable de la PRD. L’étalement général des évènements dans la PRDavo est encore plus important que dans la PRD classique. Toutefois le maximum d’énergie se situe maintenant bien sur la trace q=0 pour tous les évènements, y compris celui à 800ms, permettant de les pointer correctement.

Pour la PRDavo l’énergie correspondant à chaque échantillon du modèle \mathbf{u}_i du modèle est calculée par la formule suivante (il s’agit de l’énergie de la courbe correspondante):

$$e(\mathbf{u}_i) = \int_x (v_i + w_i x^2)^2 dx = X0v_i^2 + 2X2v_iw_i + X4w_i^2 \tag{E5.19}$$

L’étalement par la sévère sous-détermination du problème: le nombre d’inconnues (paramètres du modèle) est trop important, laissant trop de libertés pour représenter les données. Cette sous-détermination se comprend très bien si l’on considère un évènement horizontal d’amplitude constante (fig. 5.3): dans une PRD celui-ci peut en effet être expliqué par de nombreuses paraboles d’amplitudes variables qui s’inscrivent dans la largeur du signal sismique, y compris celles dont l’amplitude change de signe.

On entrevoit par conséquent ici tout l’intérêt des décompositions parcimonieuses pour forcer l’optimisation à représenter les données en utilisant le moins d’échantillons possible dans le modèle.

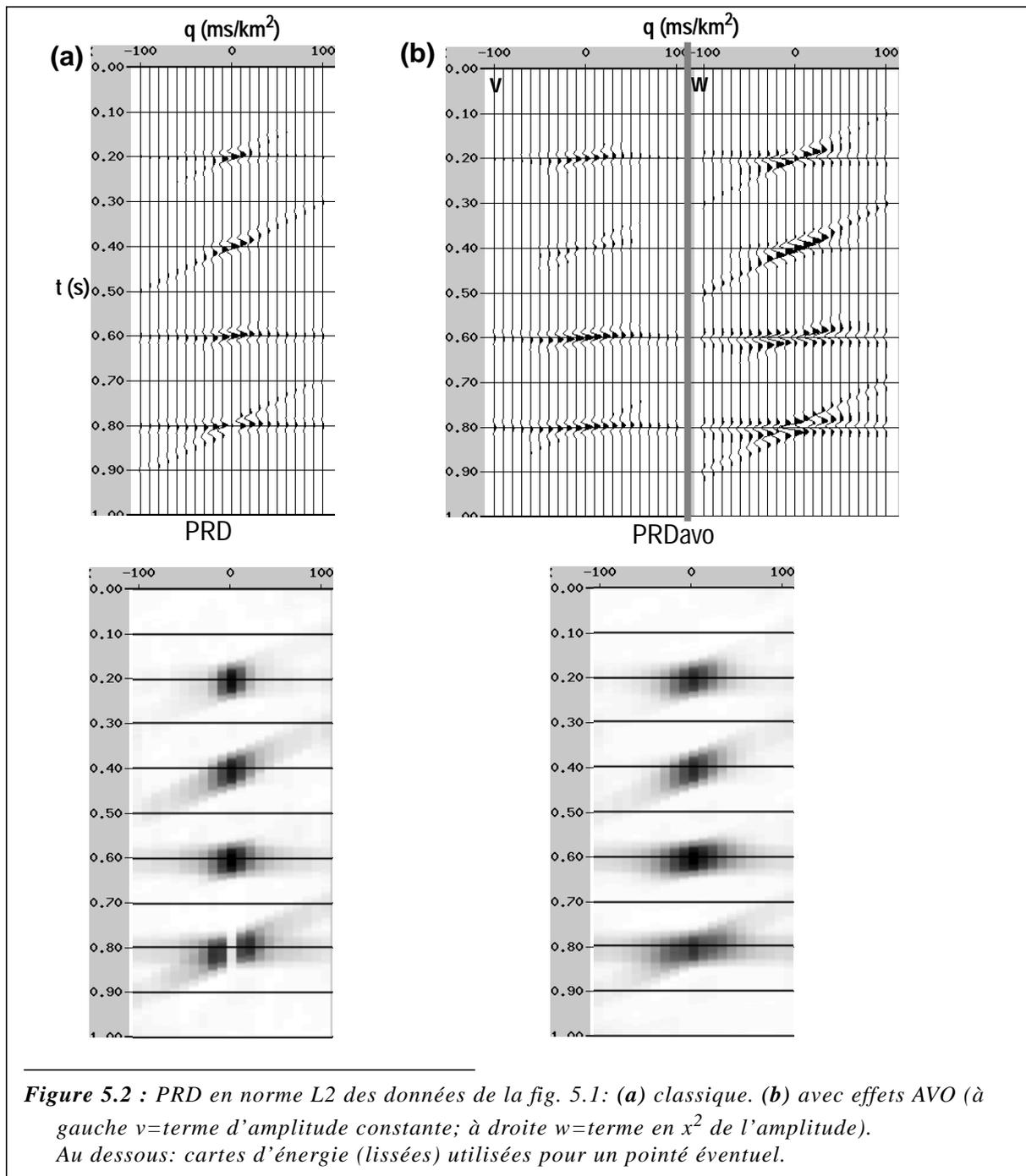


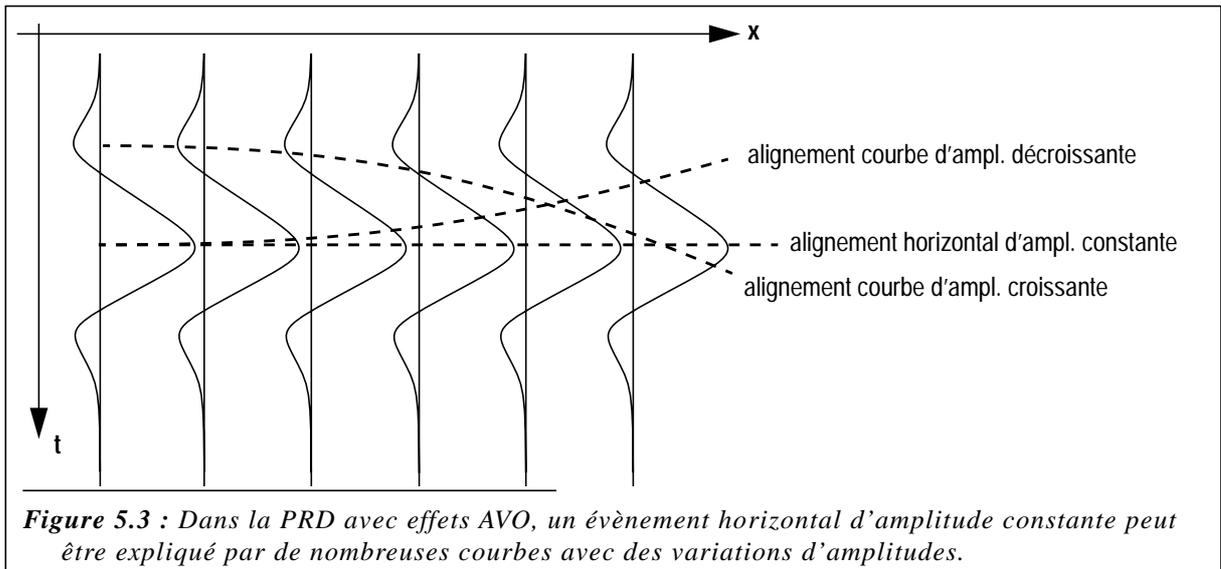
Figure 5.2 : PRD en norme L2 des données de la fig. 5.1: (a) classique. (b) avec effets AVO (à gauche v =terme d'amplitude constante; à droite w =terme en x^2 de l'amplitude). Au dessous: cartes d'énergie (lissées) utilisées pour un pointé éventuel.

5.5 - Décompositions parcimonieuses

On utilise les décompositions parcimonieuses telles que décrites dans le chapitre 4: les décompositions en norme L1 et celles calculées par analyse des rapports signal/bruit.

5.5.1 - Norme L1 sur le modèle

L'implémentation de décomposition en norme L1 dans le cas classique nécessite le calcul d'un poids sur le modèle d'après le modèle lui-même (équ. (E4.7)):



$$(\tilde{\mathbf{P}}_{Lp}(\mathbf{u}))_{ii} = \frac{p}{(|\mathbf{u}_i| + \eta)^{2-p}}$$

Le modèle a ici deux composantes. On ne voit à priori pas de raison d'appliquer un poids différent à chacune des composantes. On utilise à la place de la valeur absolue $|\mathbf{u}_i|$ l'énergie portée par chaque échantillon du modèle (équ. (E5.19)):

$$(\tilde{\mathbf{P}}_{Lp}(\mathbf{u}))_{ii} = \frac{p}{(\sqrt{e(\mathbf{u}_i)} + \eta)^{2-p}} \tag{E5.20}$$

Le poids utilisé est le même pour les deux composantes, et ne les lie pas entre elles: $\begin{bmatrix} \tilde{\mathbf{P}}_{Lp} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{P}}_{Lp} \end{bmatrix}$

5.5.2 - Décomposition par analyse des rapports Signal/Bruit

Dans l'équation (E4.30), on calcule un poids d'après les estimations des variances du bruit et des échantillons du modèle. Ces dernières pouvant être obtenue pour chaque courbe par une analyse des amplitudes le long de celle-ci (E4.31). On en tire une moyenne (qui peut servir à estimer la variance dans le modèle) et une variance (pour le bruit).

Ici, on peut essayer de retrouver pour chaque courbe le profil d'amplitude à deux coefficients par un simple ajustement. Si $y_i(x)$ est l'amplitude le long de la courbe représentée par l'échantillon i du modèle, on cherche μ_v, μ_w tels que (on omet l'indice i dans les expressions):

$$\sigma_n^2 = \min_{\mu_v, \mu_w} \left(\frac{1}{X_0} \int_x (y(x) - (\mu_v + \mu_w x^2))^2 dx \right)$$

L'expression sous l'intégrale est égale à:

$$\int_x (y(x) - (\mu_v + \mu_w x^2))^2 dx = \int_x (y(x)^2 + \mu_v^2 + \mu_w^2 x^4 - 2y(x)\mu_v - 2y(x)\mu_w x^2 + 2\mu_v \mu_w x^2) dx$$

On pose les quantités suivantes:

$$Y = \int_x y(x) dx \quad Y2 = \int_x y(x)^2 dx \quad YX2 = \int_x y(x)x^2 dx \quad (E5.21)$$

donc:

$$\int_x (y(x) - (\mu_v + \mu_w x^2))^2 dx = Y2 + X0\mu_v^2 + X4\mu_w^2 - 2Y\mu_v - 2YX2\mu_w + 2X2\mu_v\mu_w$$

On dérive cette expression par à rapport aux deux variables μ_v, μ_w :

$$\frac{\partial}{\partial \mu_v} () = 2X0\mu_v - 2Y + 2X2\mu_w$$

$$\frac{\partial}{\partial \mu_w} () = 2X4\mu_w - 2YX2 + 2X2\mu_v$$

les deux dérivées doivent être nulles au minimum, ce qui conduit à résoudre le système:

$$\begin{bmatrix} X0 & X2 \\ X2 & X4 \end{bmatrix} \begin{bmatrix} \mu_v \\ \mu_w \end{bmatrix} = \begin{bmatrix} Y \\ YX2 \end{bmatrix}$$

Soit:

$$\begin{bmatrix} \mu_v \\ \mu_w \end{bmatrix} = \frac{1}{X0X4 - X2^2} \begin{bmatrix} X4 & -X2 \\ -X2 & X0 \end{bmatrix} \begin{bmatrix} Y \\ YX2 \end{bmatrix} \quad (E5.22)$$

Pour ce minimum, la fonction vaut:

$$\begin{aligned} \sigma_n^2 &= \min_{\mu_v, \mu_w} \left(\frac{1}{X0} \int_x (y(x) - (\mu_v + \mu_w x^2))^2 dx \right) \\ &= \frac{1}{X0} (Y2 + \mu_v^2 X0 + \mu_w^2 X4 - 2\mu_v Y - 2\mu_w YX2 + 2\mu_v \mu_w X2) \\ &= \frac{1}{X0} \left(Y2 + \frac{2 X2 Y YX2 - X4 Y^2 - X0 YX2^2}{X0X4 - X2^2} \right) \end{aligned}$$

$$\sigma_n^2 = \frac{1}{X0} \left(Y2 + \frac{2 X2 Y YX2 - X4 Y^2 - X0 YX2^2}{X0X4 - X2^2} \right) \quad (E5.23)$$

Pour chaque courbe du modèle on obtient donc une estimation de la composante constante, de la composante en x^2 , et du niveau de bruit. Les deux composantes sont utilisées comme une direction de recherche linéaire pour calculer un coefficient α de mise à l'échelle de l'estimation (voir équ. (E4.35)):

$$\begin{bmatrix} \mathbf{v} \\ \mathbf{w} \end{bmatrix} = \alpha \begin{bmatrix} \mu_v \\ \mu_w \end{bmatrix} \quad (E5.24)$$

Enfin, comme on ne veut pas appliquer un poids différent sur les deux composantes, on calcule là aussi une valeur composite. Celle-ci doit être normalisée afin d'avoir une amplitude comparable aux deux composantes (on utilise l'énergie définie en (E5.19)):

$$\tilde{\sigma}_i^2 = \frac{X_0 \mathbf{v}_i^2 + 2X_2 \mathbf{v}_i \mathbf{w}_i + X_4 \mathbf{w}_i^2}{\sqrt{X_0 X_4}} = \frac{e(\mathbf{u}_i)}{\sqrt{X_0 X_4}} \quad (E5.25)$$

5.5.3 - Application (données synthétiques)

On calcule, sur les données de la figure 5.1, les PRD en norme L1 sur le modèle d'une part et par estimation des rapports S/B d'autre part. On peut comparer les décompositions classiques avec les décompositions avec effets AVO (figure 5.4). Les deux PRD classiques ne peuvent pas focaliser les événements présentant des variations d'amplitudes (seul le premier événement, d'amplitude constante, est bien focalisé). La PRD avec effets AVO calculée par estimations des rapports S/B est par contre d'excellente qualité: tous les événements sont parfaitement focalisés et, qui plus est, les rapports d'amplitudes entre les deux sous-sections représentent fidèlement les coefficients AVO des données. La focalisation pour la PRD en norme L1 est par contre assez mauvaise. Cela semble dû uniquement à un problème de convergence: la PRD a été calculée par un algorithme IRLS de type 1 (paragr. 3.5.3), avec 3 résolutions successives à poids constant (12 itérations dans chaque), ce qui suffit en général pour une PRD classique. En poussant plus loin la convergence (8 résolutions successives), on atteint la focalisation désiré (fig. 5.5).

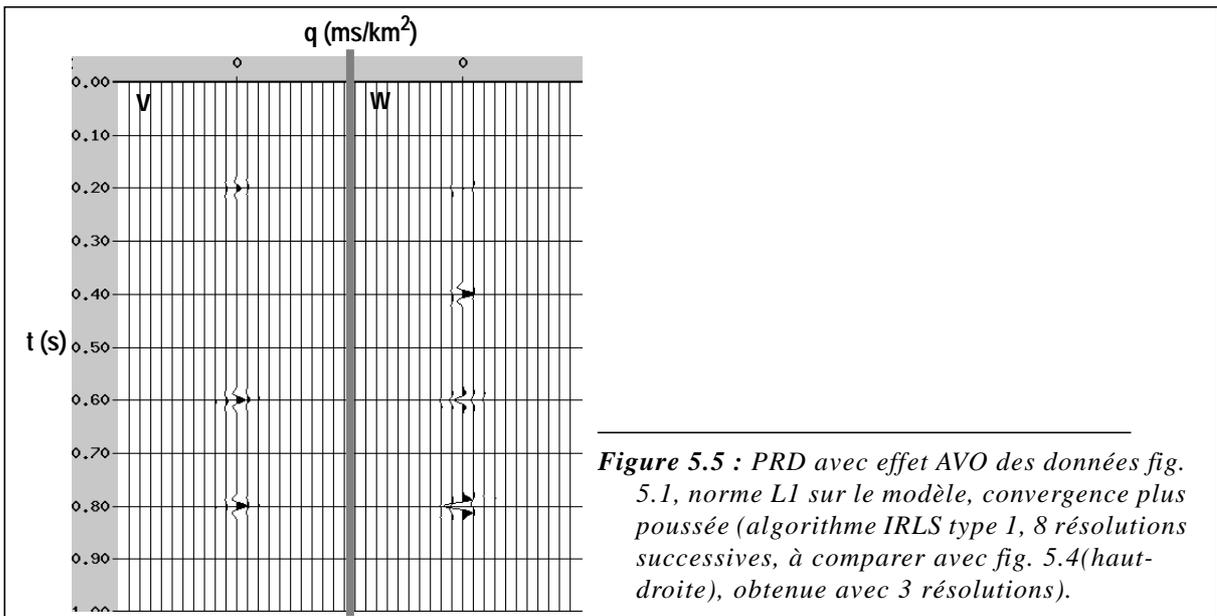
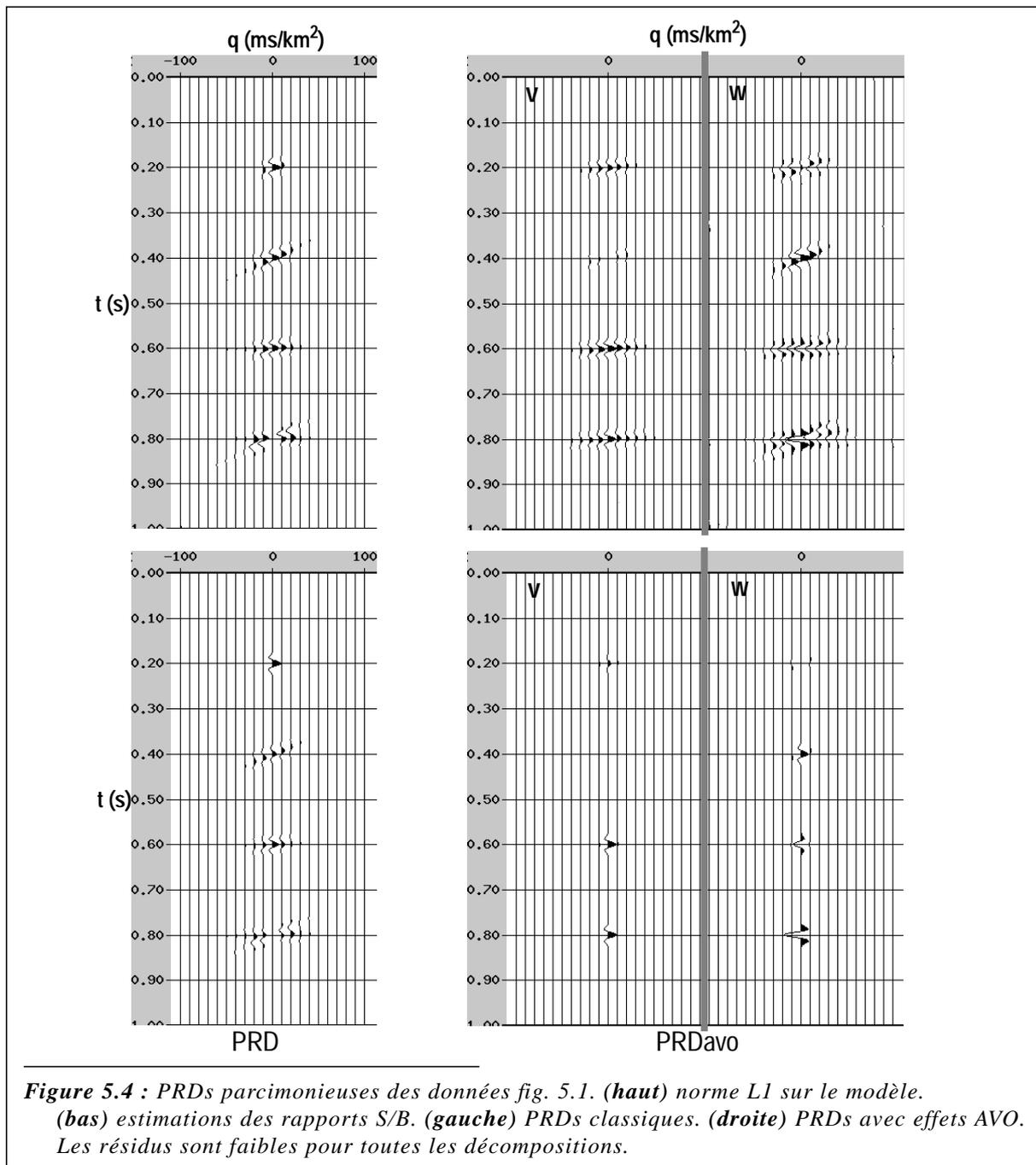


Figure 5.5 : PRD avec effet AVO des données fig. 5.1, norme L1 sur le modèle, convergence plus poussée (algorithme IRLS type 1, 8 résolutions successives, à comparer avec fig. 5.4 (haut-droite), obtenue avec 3 résolutions).

On peut sans doute expliquer ce problème de convergence pour le calcul des solutions parcimonieuses par la sous-détermination intrinsèque du problème. L'hypothèse de parcimonie est un a priori introduit pour compenser cette sous-détermination. Néanmoins, le point de départ de la décomposition en norme L1 dans l'algorithme IRLS est la décomposition en norme L2 (fig. 5.2), très éloignée de la solution souhaitée.

Le décomposition par estimation des rapports S/B semble moins sensible à ce problème de convergence, un résultat très satisfaisant étant obtenu avec 3 séries de 5 itérations (voir paragr. 4.4.2.2), paramètres habituellement choisis pour les PRD classiques. Il faut toutefois considérer que la convergence pour ce type de décomposition est très dépendante du niveau de bruit. Or ce dernier est ici très faible (on a juste ajouté un bruit gaussien dont l'écart type est de 1/10ème de l'amplitude moyenne des événements). Un bruit plus important se traduit par une bien moindre focalisation (fig. 5.6, à



comparer avec fig. 5.4(bas-droite)). Ce résultat pourrait néanmoins servir de nouvelle estimation du signal pour une nouvelle PRD par estimation des rapports S/B, afin d'améliorer la focalisation.

Toutefois les problèmes de convergences sont bien plus critiques que pour les décompositions classiques.

5.5.4 - Application (données réelles)

On teste la PRD avec effet AVO sur une collection CMP corrigée de NMO, comportant 90 traces (intertrace de 50m) (fig. 5.7). Les primaires sont approximativement horizontalisés, et on note sur certains d'entre eux de très forts effets AVO (à 2400, 2520, et 4360ms): le coefficient de réflexion croît fortement avec l'offset.

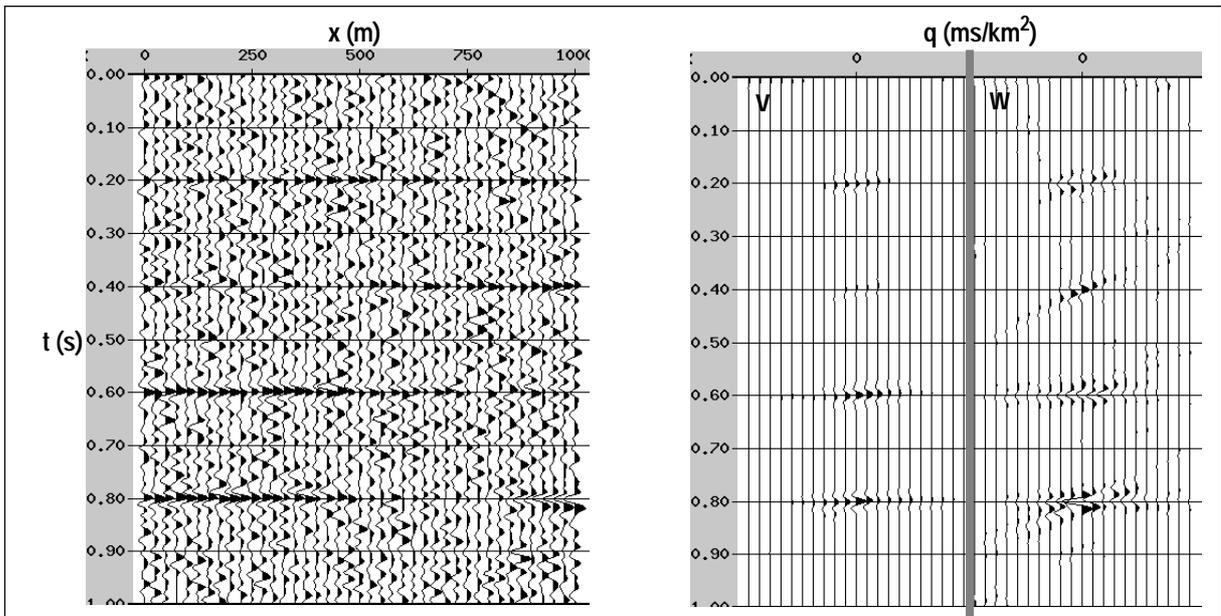


Figure 5.6 : (gauche) mêmes données que fig. 5.1, entachées d'un fort bruit gaussien. (droite) PRD avec effets AVO, par estimations des rapports S/B.

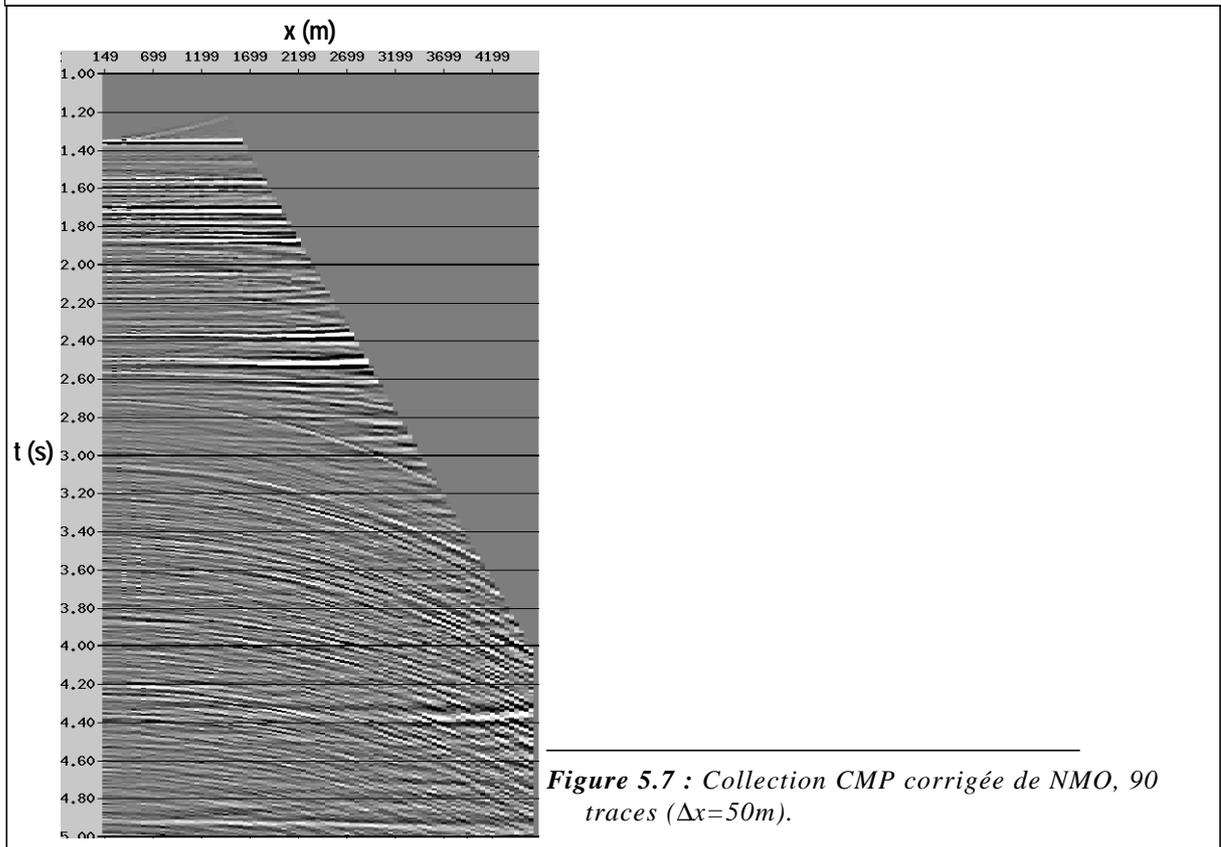


Figure 5.7 : Collection CMP corrigée de NMO, 90 traces ($\Delta x=50m$).

Sur la PRD classique (par estimation des rapports S/B) (fig. 5.8(haut)), ces événements présentant de fortes variations d'amplitude sont mal focalisés, et les résidus contiennent encore de l'énergie des plus grands offsets. En intégrant les effets AVO (fig. 5.8(bas)) on arrive à une meilleure représentation, et les résidus semblent plus uniformes (de trace à trace). Reste à savoir si des coefficients AVO peuvent être extraits de la PRD par simple comparaison des amplitudes entre les deux composantes.

On note par ailleurs quelques problèmes: l'évènement très courbé situé vers 2700ms s'écarte de l'hypothèse de 'parabolicité'. Aussi, bien qu'il soit d'amplitude quasi-constante, sa représentation dans la PRD classique est-elle étalée sur de nombreuses valeurs de q. Non seulement la PRD avec effets AVO ne résout pas le problème, mais elle l'aggrave en traitant l'évènement comme deux évènements séparés

de courbures différentes, le premier portant de l'énergie sur les offsets faibles, le second portant de l'énergie sur les grands offsets.

Bien que ce type de décomposition soit intéressante en théorie, il semble en fait que l'application sur des données réelles n'apporte pas autant que l'on pouvait en espérer. Il est par exemple très difficile de bien focaliser les événements, et la convergence du gradient conjugué est souvent délicate. Ces deux problèmes étant peut-être liés.

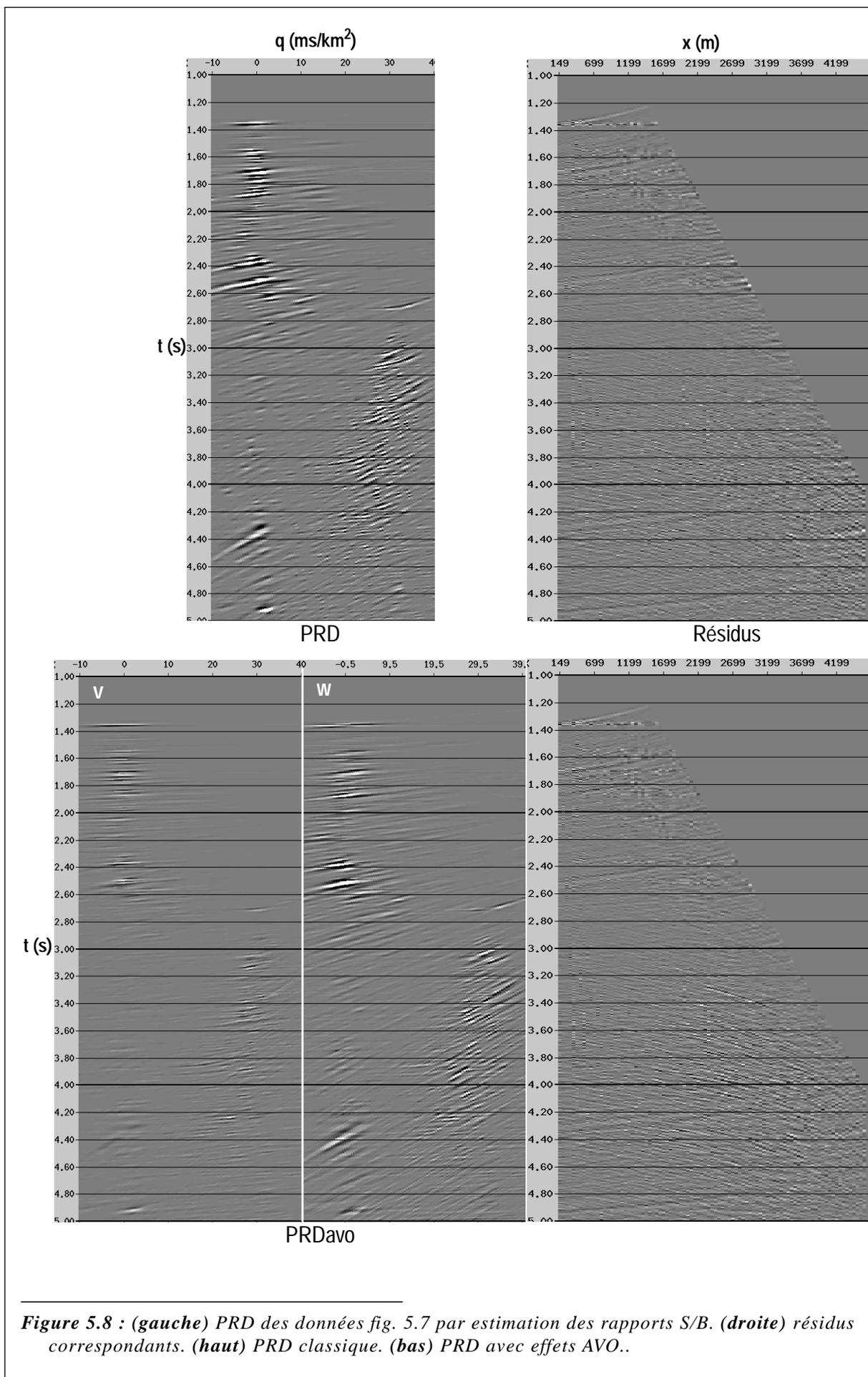


Figure 5.8 : (gauche) PRD des données fig. 5.7 par estimation des rapports S/B. (droite) résidus correspondants. (haut) PRD classique. (bas) PRD avec effets AVO..

5.6 - Conclusions

La décomposition de Radon utilisant plusieurs profils d'amplitudes pour une même famille de courbes est un problème fortement sous-déterminé. La résolution en norme L2 n'est par conséquent pas satisfaisante.

Les décompositions parcimonieuses donnent de bons résultats sur synthétiques: la focalisation obtenue est très bonne et les informations AVO pourrait être obtenues par un simple pointé. Sur données réelles en revanche la focalisation atteinte n'est pas aussi bonne. Néanmoins, l'extraction des coefficients AVO devrait être possible.

On ne peut en fait pas résoudre simultanément de manière satisfaisante la séparation d'évènements interférents et la détermination de leurs profils d'amplitudes.

Dans tous les cas, la convergence des optimisations est relativement lente comparée aux décompositions classique. La sous-détermination du problème en est sans doute la cause. Il ne faudrait peut-être utiliser les amplitudes variables qu'en certains points du modèle, qui correspondraient à des pointés d'évènements déjà repérés.

6 - (τ - p) 3D

6.1 - Introduction

On étudie dans ce chapitre l'extension 3D (2 variables spatiales) de la Décomposition de Radon Linéaire.

Conceptuellement, l'ajout de dimensions ne pose pas de nouveau problème théorique. Dès l'origine Radon (1917) définit sa transformée dans des espaces de dimensions quelconques (voir aussi Helgason, 1980).

Les implémentations peuvent par contre varier notablement. Ainsi, si les implémentations en domaine temps peuvent être identiques -car n'exploitant pas de simplifications particulières-, celles dans le domaine de Fourier peuvent différer: la LRD 2D en norme L2 aboutit à la résolution de plusieurs systèmes de Toeplitz., alors que pour la LRD 3D les simplifications de ce type dépendent de la géométrie des données (une grille régulière permet de décomposer le problème 3D en 2x2D, alors qu'une grille irrégulière nécessite la résolution de plusieurs systèmes 'Toeplitz par blocs').

Les problèmes d'échantillonnage et d'aliasing peuvent être directement extrapolés du cas 2D.

Les applications trouvées dans la littérature concernent le filtrage du ground-roll (Donati et al., 1996) et l'interpolation de traces ou la régularisation d'acquisitions (Shin and Sumner, 1994; Hindriks et al., 1997) -sujet étudié dans le chapitre suivant-, et utilisent toutes les décompositions en norme L2.

Les principes et les implémentations des décompositions parcimonieuses définis pour le cas 2D sont applicables directement au cas 3D (l'hypothèse de parcimonie est d'ailleurs d'autant plus justifiée dans un espace de dimension élevée).

6.2 - Définitions

6.2.1 - Variables continues

La décomposition de Radon 3D $(\tau, p1, p2)$ est définie de manière analogue à la décomposition 2D. Pour des variables continues, les supports des données et des modèles sont:

$$D = \{(t, x1, x2); T_i(x1, x2) \leq t \leq T_f(x1, x2) ; x1_{min} \leq x1 \leq x1_{max}; x2_{min} \leq x2 \leq x2_{max}\} \quad (E6.1)$$

$$U = \{(\tau, p1, p2); \tau_i(p1, p2) \leq \tau \leq \tau_f(p1, p2) ; p1_{min} \leq p1 \leq p1_{max}; p2_{min} \leq p2 \leq p2_{max}\} \quad (E6.2)$$

Les données et les modèles sont des fonctions de carrés intégrales sur D et U.

L'application directe, ou modélisation, est définie comme dans le cas 2D (E1.21) par une superposition de plans (au lieu de droites) d'amplitudes constantes:

$$\begin{aligned} \underline{\mathbf{L}}: \quad \text{IU} &\rightarrow \text{ID} \\ \underline{\mathbf{L}}\mathbf{u}(t, x_1, x_2) &= \int_{p_1} \int_{p_2} \int_{\tau} \mathbf{u}(\tau, p_1, p_2) \delta(t - (\tau + x_1 p_1 + x_2 p_2)) d\tau dp_1 dp_2 \\ &= \int_{p_1} \int_{p_2} \mathbf{u}(t - x_1 p_1 + x_2 p_2, p_1, p_2) dp_1 dp_2 \end{aligned} \quad (\text{E6.3})$$

L'opérateur de stack, de l'espace des données vers l'espace des modèles, est l'adjoint de l'opérateur de modélisation pour le produit scalaire usuel:

$$\begin{aligned} \underline{\mathbf{L}}^*: \quad \text{ID} &\rightarrow \text{IU} \\ \underline{\mathbf{L}}^*\mathbf{d}(\tau, p_1, p_2) &= \int_{x_1} \int_{x_2} \mathbf{d}(\tau + x_1 p_1 + x_2 p_2, x_1, x_2) dx_1 dx_2 \end{aligned} \quad (\text{E6.4})$$

6.2.2 - Domaine de Fourier

Les formulations fréquentielles des opérateurs sont:

$$\begin{aligned} \underline{\mathbf{L}}\hat{\mathbf{u}}(\omega, x_1, x_2) &= \int_{p_1} \int_{p_2} \hat{\mathbf{u}}(\omega, p_1, p_2) e^{-i\omega(x_1 p_1 + x_2 p_2)} dp_1 dp_2 \\ \underline{\mathbf{L}}^*\hat{\mathbf{d}}(\tau, p_1, p_2) &= \int_{x_1} \int_{x_2} \hat{\mathbf{d}}(\omega, x_1, x_2) e^{i\omega(x_1 p_1 + x_2 p_2)} dx_1 dx_2 \end{aligned} \quad (\text{E6.5})$$

L'opérateur composé montre que le problème 3D peut se décomposer en '2x2D':

$$\begin{aligned} &\underline{\mathbf{L}}^*\underline{\mathbf{L}}\hat{\mathbf{u}}(\omega, p_1, p_2) \\ &= \int_{x_1} \int_{x_2} \int_{p_1'} \int_{p_2'} \hat{\mathbf{u}}(\omega, p_1', p_2') e^{i\omega(x_1(p_1 - p_1') + x_2(p_2 - p_2'))} dp_1' dp_2' dx_1 dx_2 \\ &= \int_{p_1'} \int_{p_2'} \hat{\mathbf{u}}(\omega, p_1', p_2') \int_{x_1} e^{i\omega x_1(p_1 - p_1')} dx_1 \int_{x_2} e^{i\omega x_2(p_2 - p_2')} dx_2 dp_1' dp_2' \end{aligned}$$

On reconnaît dans les intégrales simples les opérateurs 2D de la LRD définis en (E1.52):

$$\begin{aligned} \underline{\mathbf{L}}^* \underline{\mathbf{L}} \hat{\mathbf{u}}(\omega, p_1, p_2) &= \int_{p_1'} \int_{p_2'} \hat{\mathbf{u}}(\omega, p_1', p_2') \sigma_{x_1}(\omega, p_1 - p_1') \sigma_{x_2}(\omega, p_2 - p_2') dp_1' dp_2' \\ &= \int_{p_1'} (\hat{\mathbf{u}}(\omega, p_1', p_2') *_{p_2} \sigma_{x_2}(\omega, p_2)) \sigma_{x_1}(\omega, p_1 - p_1') dp_1' \end{aligned}$$

$$\boxed{\underline{\mathbf{L}}^* \underline{\mathbf{L}} \hat{\mathbf{u}}(\omega, p_1, p_2) = (\hat{\mathbf{u}}(\omega, p_1', p_2') *_{p_2} \sigma_{x_2}(\omega, p_2)) *_{p_1} \sigma_{x_1}(\omega, p_1)} \quad (\text{E6.6})$$

L'opérateur composé s'exprime donc comme deux convolutions 1D successives suivant chacune des variables paramétrisant les pentes dans les deux directions spatiales.

On ne peut aboutir à cette écriture que si les bornes de chacune des deux variables spatiales (x_1 et x_2) ne dépendent pas de l'autre variable (autrement dit si le support des données est rectangulaire, éventuellement infini). Dans le cas contraire, on ne peut pas séparer les deux variables spatiales dans les intégrales, et on conserverait un opérateur de convolution 2D.

6.2.3 - Variables discrètes

6.2.3.1 - Grille des données quelconque

Supposons que les données soient constituées de N_x traces, réparties spatialement d'une manière quelconque, de coordonnées $(x_{1_1}, x_{2_1}), (x_{1_2}, x_{2_2}), \dots, (x_{1_{N_x}}, x_{2_{N_x}})$.

$$\mathbf{d} = (d_1, d_2, d_3, \dots, \dots, \dots, d_{N_x})^t \quad (\text{E6.7})$$

Le modèle est un vecteur de $N_{p1} \cdot N_{p2}$ éléments, distribués sur une **grille rectangulaire régulière** (voir fig. 6.1):

$$\begin{aligned} \mathbf{u} &= (u_{1,1}, u_{2,1}, \dots, u_{N_{p1},1}, u_{1,2}, u_{2,2}, \dots, \dots, \dots, u_{1,N_{p2}}, u_{2,N_{p2}}, \dots, u_{N_{p1},N_{p2}})^t \\ p_{1,i,j} &= p_{1_i} = p_{1_{\min}} + (i-1)\Delta p_1 \\ p_{2,i,j} &= p_{2_j} = p_{2_{\min}} + (j-1)\Delta p_2 \end{aligned} \quad (\text{E6.8})$$

La matrice exprimant la transformée directe est:

$$\underline{\mathbf{L}} = (\underline{\mathbf{K}}_1, \underline{\mathbf{K}}_2, \dots, \underline{\mathbf{K}}_{N_{p2}}) \quad (\text{E6.9})$$

Où les matrices $\underline{\mathbf{K}}_j$ sont de dimension $N_x \cdot N_{p1}$:

$$\underline{\mathbf{K}}_j = \begin{bmatrix} e^{-i\omega(x_{1_1}p_{1_1} + x_{2_1}p_{2_j})} & e^{-i\omega(x_{1_1}p_{1_2} + x_{2_1}p_{2_j})} & \dots & e^{-i\omega(x_{1_1}p_{1_{N_{p1}}} + x_{2_1}p_{2_j})} \\ e^{-i\omega(x_{1_2}p_{1_1} + x_{2_2}p_{2_j})} & e^{-i\omega(x_{1_2}p_{1_2} + x_{2_2}p_{2_j})} & \dots & e^{-i\omega(x_{1_2}p_{1_{N_{p1}}} + x_{2_2}p_{2_j})} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ e^{-i\omega(x_{1_{N_x}}p_{1_1} + x_{2_{N_x}}p_{2_j})} & e^{-i\omega(x_{1_{N_x}}p_{1_2} + x_{2_{N_x}}p_{2_j})} & \dots & e^{-i\omega(x_{1_{N_x}}p_{1_{N_{p1}}} + x_{2_{N_x}}p_{2_j})} \end{bmatrix} \quad (\text{E6.10})$$

L'opérateur composé s'écrit:

$$\underline{\underline{L}}^* \underline{\underline{L}} = \begin{bmatrix} \underline{\underline{K}}_1^* \underline{\underline{K}}_1 & \underline{\underline{K}}_1^* \underline{\underline{K}}_2 & \dots & \underline{\underline{K}}_1^* \underline{\underline{K}}_{Np2} \\ \underline{\underline{K}}_2^* \underline{\underline{K}}_1 & \underline{\underline{K}}_2^* \underline{\underline{K}}_2 & \dots & \underline{\underline{K}}_2^* \underline{\underline{K}}_{Np2} \\ \dots & \dots & \dots & \dots \\ \underline{\underline{K}}_{Np2}^* \underline{\underline{K}}_1 & \underline{\underline{K}}_{Np2}^* \underline{\underline{K}}_2 & \dots & \underline{\underline{K}}_{Np2}^* \underline{\underline{K}}_{Np2} \end{bmatrix} \quad (E6.11)$$

Les matrices $\underline{\underline{K}}_j^* \underline{\underline{K}}_k$ sont carrées de dimension $Np1.Np1$ et leurs éléments s'écrivent:

$$\begin{aligned} (\underline{\underline{K}}_j^* \underline{\underline{K}}_k)_{m,n} &= \sum_{s=1}^{Nx} e^{i\omega(x1_s p1_m + x2_s p2_j)} e^{-i\omega(x1_s p1_n + x2_s p2_k)} \\ &= \sum_{s=1}^{Nx} e^{i\omega(x1_s (p1_m - p1_n) + x2_s (p2_j - p2_k))} \\ &= \sum_{s=1}^{Nx} e^{i\omega(x1_s (m-n)\Delta p1 + x2_s (j-k)\Delta p2)} \end{aligned} \quad (E6.12)$$

Propriétés des matrices $\underline{\underline{K}}_j^* \underline{\underline{K}}_k$:

Elles ont une structure de Toeplitz ($(\underline{\underline{K}}_j^* \underline{\underline{K}}_k)_{m+h, n+h} = (\underline{\underline{K}}_j^* \underline{\underline{K}}_k)_{m,n}$), mais ne sont pas hermitiennes, sauf pour $k=j$ ($(\underline{\underline{K}}_j^* \underline{\underline{K}}_j)_{m,n} = (\underline{\underline{K}}_j^* \underline{\underline{K}}_j)_{n,m}^*$).

On a aussi pour tout h : $\underline{\underline{K}}_{j+h}^* \underline{\underline{K}}_{k+h} = \underline{\underline{K}}_j^* \underline{\underline{K}}_k$, c'est à dire que $\underline{\underline{L}}^* \underline{\underline{L}}$ possède une structure de **Toeplitz 'par blocs'**.

Posons $\underline{\underline{K}}_j^* \underline{\underline{K}}_k = \underline{\underline{Q}}_{j-k}$. On vérifie facilement que $\underline{\underline{Q}}_{-j} = \underline{\underline{Q}}_j^*$. On peut donc écrire:

$$\underline{\underline{L}}^* \underline{\underline{L}} = \begin{bmatrix} \underline{\underline{Q}}_0 & \underline{\underline{Q}}_1^* & \dots & \dots & \underline{\underline{Q}}_{Np2-1}^* \\ \underline{\underline{Q}}_1 & \underline{\underline{Q}}_0 & \underline{\underline{Q}}_1^* & & \\ \dots & \underline{\underline{Q}}_1 & & & \\ \dots & & & & \underline{\underline{Q}}_1^* \\ \underline{\underline{Q}}_{Np2-1} & & & \underline{\underline{Q}}_1 & \underline{\underline{Q}}_0 \end{bmatrix} \quad (E6.13)$$

Les éléments des matrices $\underline{\underline{Q}}_j$ étant, d'après (E6.12):

$$(\underline{\underline{Q}}_j)_{m,n} = \sum_{s=1}^{Nx} e^{i\omega(x1_s (m-n)\Delta p1 + x2_s j\Delta p2)} \quad (E6.14)$$

6.2.3.2 - Grille des données ‘pseudo-régulière’

La grille des données est ‘pseudo-régulière’ si les coordonnées des traces peuvent se mettre sous la forme:

$$\begin{aligned}
 &(x1_1, x2_1), (x1_2, x2_1), \dots, (x1_{Nx1}, x2_1) \\
 &(x1_1, x2_2), (x1_2, x2_2), \dots, (x1_{Nx1}, x2_2) \\
 &\dots \\
 &(x1_1, x2_{Nx2}), (x1_2, x2_{Nx2}), \dots, (x1_{Nx1}, x2_{Nx2})
 \end{aligned}
 \quad \text{avec } N=Nx1.Nx2$$

Le terme ‘pseudo’ se justifie car on n’impose pas un pas d’échantillonnage constant pour ces variables (voir fig. 6.1).

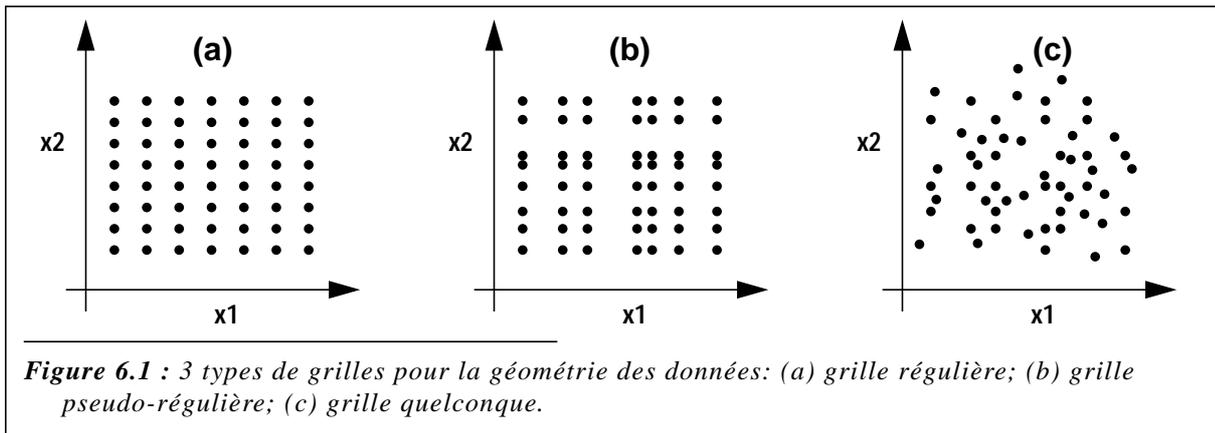


Figure 6.1 : 3 types de grilles pour la géométrie des données: (a) grille régulière; (b) grille pseudo-régulière; (c) grille quelconque.

On peut maintenant séparer les contributions des 2 variables spatiales dans (E6.14):

$$\begin{aligned}
 (\underline{Q}_j)_{m,n} &= \sum_{s=1}^{Nx1} \sum_{r=1}^{Nx2} e^{i\omega(x1_s(m-n)\Delta p1 + x2_r j \Delta p2)} \\
 &= \left(\sum_{r=1}^{Nx2} e^{i\omega x2_r j \Delta p2} \right) \left(\sum_{s=1}^{Nx1} e^{i\omega x1_s(m-n)\Delta p1} \right) \\
 &= \Sigma_{x2}(\omega, j) \Sigma_{x1}(\omega, m-n)
 \end{aligned}
 \tag{E6.15}$$

On reconnaît en effet les opérateurs de convolution de la LRD sous leur forme discrète, définis dans l’équation (E1.61):

$$\begin{aligned}
 \underline{Q}_j &= \Sigma_{x2}(\omega, j) \underline{Q}_{x1} \\
 \underline{Q}_{x1} &= \begin{bmatrix} \Sigma_{x1}(\omega, 0) & \Sigma_{x1}(\omega, 1)^* & \dots & \dots & \Sigma_{x1}(\omega, Np1-1)^* \\ \Sigma_{x1}(\omega, 1) & \Sigma_{x1}(\omega, 0) & & & \dots \\ \dots & & & & \dots \\ \dots & & & & \Sigma_{x1}(\omega, 1)^* \\ \Sigma_{x1}(\omega, Np1-1) & \dots & \dots & \Sigma_{x1}(\omega, 1) & \Sigma_{x1}(\omega, 0) \end{bmatrix}
 \end{aligned}
 \tag{E6.16}$$

Par conséquent, la matrice $\tilde{\mathbf{L}}^* \tilde{\mathbf{L}}$ peut se décomposer simplement (on omet la variable ω dans la suite des notations):

$$\tilde{\mathbf{L}}^* \tilde{\mathbf{L}} = \begin{bmatrix} \Sigma_{x_2}(0) \mathbf{Q}_{x_1} & \Sigma_{x_2}(1)^* \mathbf{Q}_{x_1} & \dots & \Sigma_{x_2}(N_{p2}-1)^* \mathbf{Q}_{x_1} \\ \Sigma_{x_2}(1) \mathbf{Q}_{x_1} & & & \dots \\ \dots & & & \Sigma_{x_2}(1)^* \mathbf{Q}_{x_1} \\ \Sigma_{x_2}(N_{p2}-1) \mathbf{Q}_{x_1} & \dots & \Sigma_{x_2}(1) \mathbf{Q}_{x_1} & \Sigma_{x_2}(0) \mathbf{Q}_{x_1} \end{bmatrix} \quad (\text{E6.17})$$

Cette matrice peut s'écrire de manière compacte en utilisant des produits de Kronecker (voir nomenclature):

$$\begin{aligned} \tilde{\mathbf{L}}^* \tilde{\mathbf{L}} &= \mathbf{Q}_{x_1} \otimes \mathbf{Q}_{x_2} \\ &= (\mathbf{Q}_{x_1} \otimes \mathbf{Id}_{N_{p2}})(\mathbf{Id}_{N_{p1}} \otimes \mathbf{Q}_{x_2}) \end{aligned} \quad (\text{E6.18})$$

Cette formulation montre que l'on peut résoudre le problème inverse en deux temps, chacun correspondant à une des deux directions spatiales. D'ailleurs, les matrices \mathbf{Q}_{x_1} et \mathbf{Q}_{x_2} ne sont rien d'autre que les opérateurs composés $\tilde{\mathbf{L}}^* \tilde{\mathbf{L}}$ de la LRD 2D par rapport aux variables x_1 et x_2 respectivement (équ. (E1.61)).

6.3 - Echantillonnage et aliasing

L'expression 2 x 2D des opérateurs de la LRD 3D (équ. (E6.6)) montrent que les règles d'échantillonnages pour toutes les variables du problème sont les mêmes que celles de la LRD 2D (équ. (E2.1), (E2.5)) pour chacune des directions spatiales prises indépendamment.

En découplant ainsi les deux variables spatiales, on se limite toutefois à considérer des grilles rectangulaires (voir fig. 6.1(b)). Cela peut conduire à des échantillonnages un peu trop luxueux dans certains cas: si par exemple le domaine de pentes (p_1, p_2) des données est circulaire, on peut montrer que la grille régulière optimum (qui nécessite le moins de points d'échantillonnage) est une grille hexagonale (Bardan, 1997). Une grille rectangulaire de pas $(\Delta x_1, \Delta x_2)$ permet en fait d'échantillonner correctement des données dont le domaine de pentes (p_1, p_2) est inscrit dans rectangle de dimensions $\left(\frac{1}{\omega \Delta x_1}, \frac{1}{\omega \Delta x_2} \right)$.

D'une manière plus générale, une grille régulière quelconque peut se définir -de manière non unique- par deux vecteurs \mathbf{v} et \mathbf{w} (fig. 6.2).

La grille définie par la base (\mathbf{v}, \mathbf{w}) correspond à la fonction d'échantillonnage formée de diracs aux noeuds de la grille:

$$E(\mathbf{x}) = \sum_{r \in \mathbb{IN}} \sum_{s \in \mathbb{IN}} \delta(\mathbf{x} - (r\mathbf{v} + s\mathbf{w})) \quad (\text{E6.19})$$

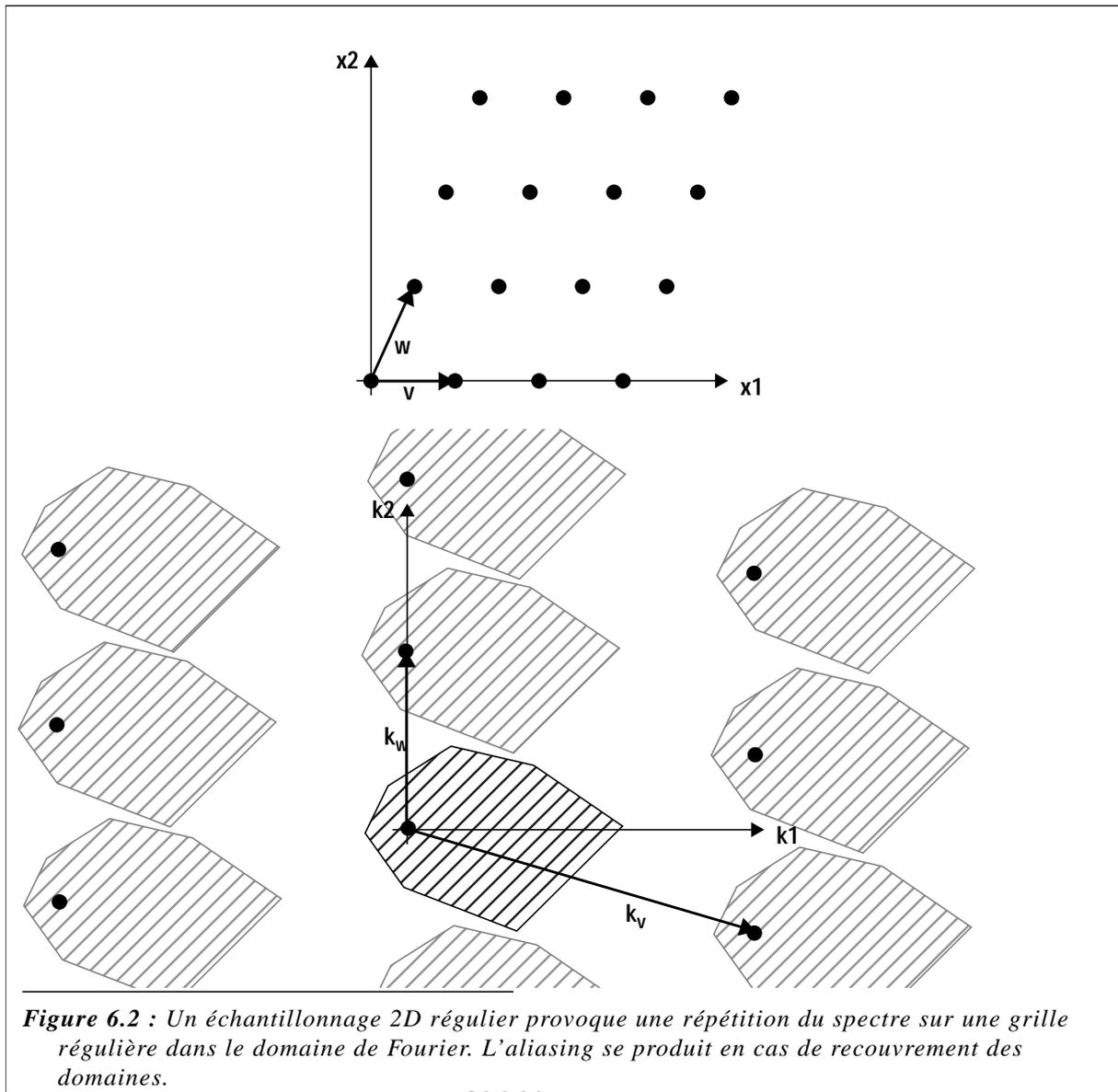


Figure 6.2 : Un échantillonnage 2D régulier provoque une répétition du spectre sur une grille régulière dans le domaine de Fourier. L'aliasing se produit en cas de recouvrement des domaines.

La transformée de Fourier (en nombre d'onde k) de cette fonction d'échantillonnage est:

$$FE(\mathbf{k}) = \iint_{\mathbf{x}} \sum_r \sum_s \delta(\mathbf{x} - (r\mathbf{v} + s\mathbf{w})) e^{-i(\mathbf{k} \cdot \mathbf{x})} d\mathbf{x} \tag{E6.20}$$

On obtient en supprimant le dirac dans l'intégrale de surface:

$$FE(\mathbf{k}) = \sum_r \sum_s e^{-i(\mathbf{k} \cdot (r\mathbf{v} + s\mathbf{w}))}$$

Cette fonction est non nulle (et prend la valeur d'un dirac) si $\mathbf{k} \cdot \mathbf{v}$ et $\mathbf{k} \cdot \mathbf{w}$ sont des multiples entiers de 2π . On définit donc les vecteurs \mathbf{k}_v et \mathbf{k}_w par les relations:

$$\begin{aligned} \mathbf{v} \cdot \mathbf{k}_v &= \mathbf{w} \cdot \mathbf{k}_w = 2\pi \\ \mathbf{v} \cdot \mathbf{k}_w &= \mathbf{w} \cdot \mathbf{k}_v = 0 \end{aligned} \tag{E6.21}$$

Le spectre de la fonction d'échantillonnage est alors égal à:

$$FE(\mathbf{k}) = 2\pi \sum_m \sum_n \delta(\mathbf{k} - (m\mathbf{k}_v + n\mathbf{k}_w)) \quad (\text{E6.22})$$

Les vecteurs \mathbf{k}_v et \mathbf{k}_w définissent également une grille régulière, qui traduit la périodicisation du spectre due à la discrétisation (fig. 6.2). Il ne doit pas y avoir de recouvrement entre le support initial du spectre des données et ses répétitions périodiques. Notons par ailleurs que la forme de ce support peut être quelconque, un support rectangulaire tel qu'on le considère habituellement n'étant qu'un cas particulier.

Le passage en pentes est immédiat, le rapport entre le nombre d'onde spatial et la pente étant égal à la pulsation temporelle: $\mathbf{k} = \omega\mathbf{p}$

A titre d'exemple, dans un dispositif d'acquisition marine 2D avec une distance entre tirs Δx égale à la distance entre récepteurs, on aboutit dans le domaine point milieu/offset à des géométries de collections point milieu périodiques par groupe de 2 CMP (fig. 6.3). On peut choisir par exemple comme vecteurs \mathbf{v} et \mathbf{w} :

$$\mathbf{v} = \left(\frac{\Delta x}{2}, \Delta x \right) \quad \mathbf{w} = (0, 2\Delta x)$$

On en déduit les vecteurs \mathbf{p}_v et \mathbf{p}_w par les relations (E6.21):

$$\mathbf{p}_v = \left(\frac{4\pi}{\omega\Delta x}, 0 \right) \quad \mathbf{p}_w = \left(-\frac{2\pi}{\omega\Delta x}, \frac{\pi}{\omega\Delta x} \right)$$

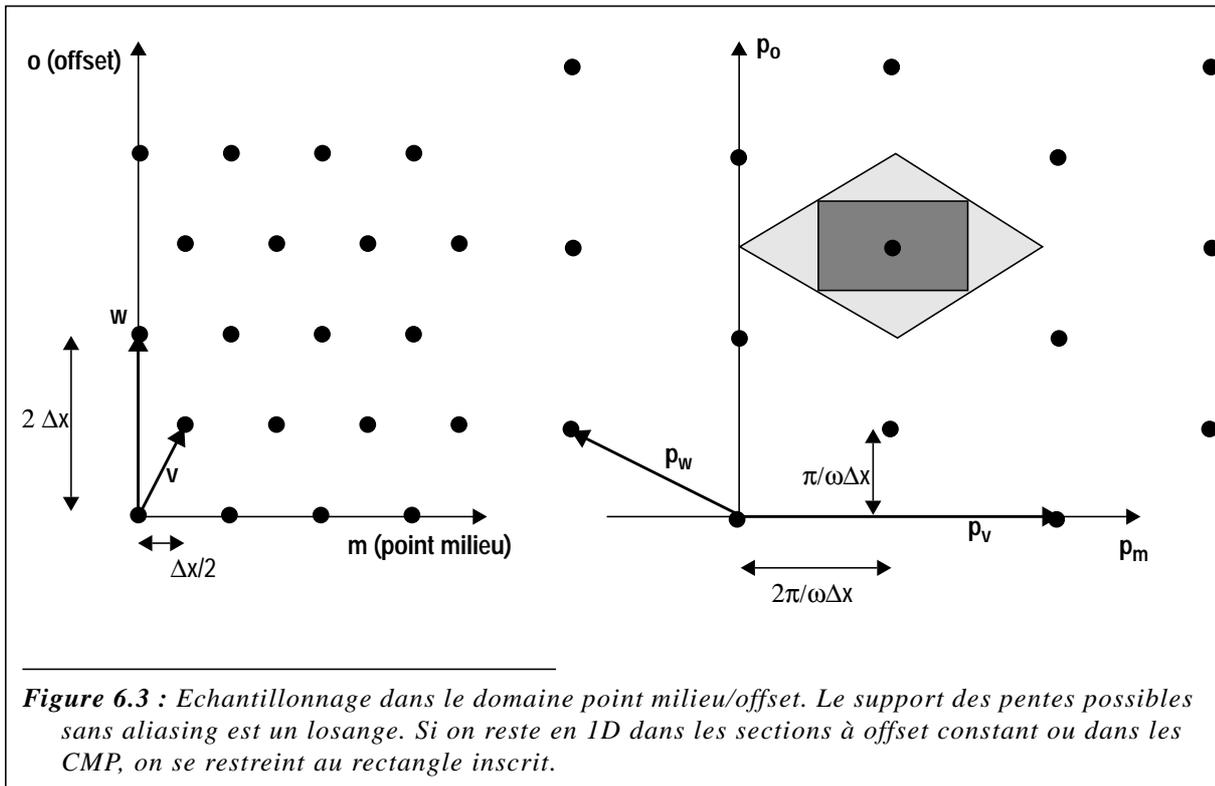
On peut donc échantillonner correctement avec cette grille des données dont les pentes suivant l'abscisse point milieu (p_m) et l'offset (p_o) sont inscrites dans un losange (fig. 6.3). Lorsque l'on travaille dans les sections CMP ou à offset constant, on se limite à des gammes possibles pour les pentes par rapport à l'échantillonnage 1D dans ces sections (Δx en iso-offset, $2\Delta x$ en iso-point milieu) d'après (E2.5): $2\pi/\omega\Delta x$ pour p_m , $\pi/\omega\Delta x$ pour p_o . Cela correspond à un rectangle inscrit dans le losange, soit un domaine possible plus réduit. En pratique cela signifie que l'on peut se retrouver artificiellement aliéné en travaillant dans certains sous-espaces alors que l'échantillonnage 3D est suffisant. On a donc tout intérêt à ré-échantillonner dans l'espace 3D avant de passer dans ces sous-espaces (en doublant le nombre de traces dans chaque point milieu par exemple).

6.4 - Décomposition en norme L2

L'inversion en moindres carrés revient à inverser l'opérateur composé $\tilde{\mathbf{L}}^*\tilde{\mathbf{L}}$ -éventuellement augmenté d'un facteur d'amortissement $\varepsilon\mathbf{Id}$ -, ou à résoudre le système linéaire:

$$\tilde{\mathbf{L}}^*\mathbf{d} = (\tilde{\mathbf{L}}^*\tilde{\mathbf{L}} + \varepsilon\mathbf{Id})\mathbf{u}$$

Dans le domaine temps, on résout le système itérativement par un gradient conjugué, préconditionné par un filtre rho 3D. Dans le domaine de Fourier, on peut calculer une solution exacte pour chaque fréquence, les résolutions des systèmes linéaires exploitant dans ce cas les structures particulières de la matrice $\tilde{\mathbf{L}}^*\tilde{\mathbf{L}}$.



6.4.1 - Domaine de Fourier, grille des données quelconque

La matrice $\mathbf{L}^*\mathbf{L}$ possède pour chaque fréquence une structure Toeplitz par bloc, chaque bloc possédant lui-même une structure de Toeplitz (équ. (E6.13) et (E6.14)).

On peut exploiter la structure de Toeplitz par blocs par une adaptation de l'algorithme de Levinson, conçu pour la résolution des systèmes de Toeplitz, en substituant des matrices carrées à la place des scalaires (voir le détail de l'algorithme dans l'annexe C). Malheureusement, avec cette méthode la structure de Toeplitz interne dans chacun des blocs n'est pas exploitée.

La résolution d'un système de Toeplitz classique de dimension n réclame de l'ordre de n^2 opérations, au lieu des n^3 nécessaires pour un système quelconque, ce qui représente un gain considérable. Ici, la dimension du système est $(Np_1.Np_2)$, qui nécessiterait de l'ordre de $(Np_1.Np_2)^3$ opérations. Avec l'algorithme de Levinson adapté, on arrive à $(Np_1^3.Np_2^2)$: on a le même gain qu'avec l'algorithme classique, mais seulement par rapport au nombre de blocs (Np_2) de la matrice. Il est clair que l'avantage obtenu est faible: si on prend par exemple $Np_1=Np_2$ et $Np=Np_1.Np_2$, on passe de Np^3 à $Np^{2.5}$. Quand Np_1 et Np_2 sont différents, on intéresse à organiser le modèle de telle sorte à ce que le nombre de blocs de $\mathbf{L}^*\mathbf{L}$ soit égal au plus grand de ces deux nombres (les directions p_1 et p_2 jouent des rôles symétriques et sont interchangeables).

Rappelons que le stack réclame de l'ordre de Np^2 opération (dans une configuration courante où $Np=Nx$). L'inversion représente donc une part croissante des calculs quand la taille du système augmente.

En tout état de cause, cette méthode est très lente et inexploitable dans la pratique. Une résolution non exacte par gradient conjugué serait certainement plus rapide, mais dans ce cas on perdrait le principal avantage à travailler dans le domaine de Fourier, à savoir obtenir une solution exacte par inversion directe d'un petit système linéaire. Quitte à travailler par gradient conjugué, autant le faire directement dans le domaine temporel.

A titre d'exemple, on effectue la décomposition d'un petit jeu de données de 121 traces ($N_{x1}=N_{x2}=11$), contenant 4 évènements plans (figure 6.4).

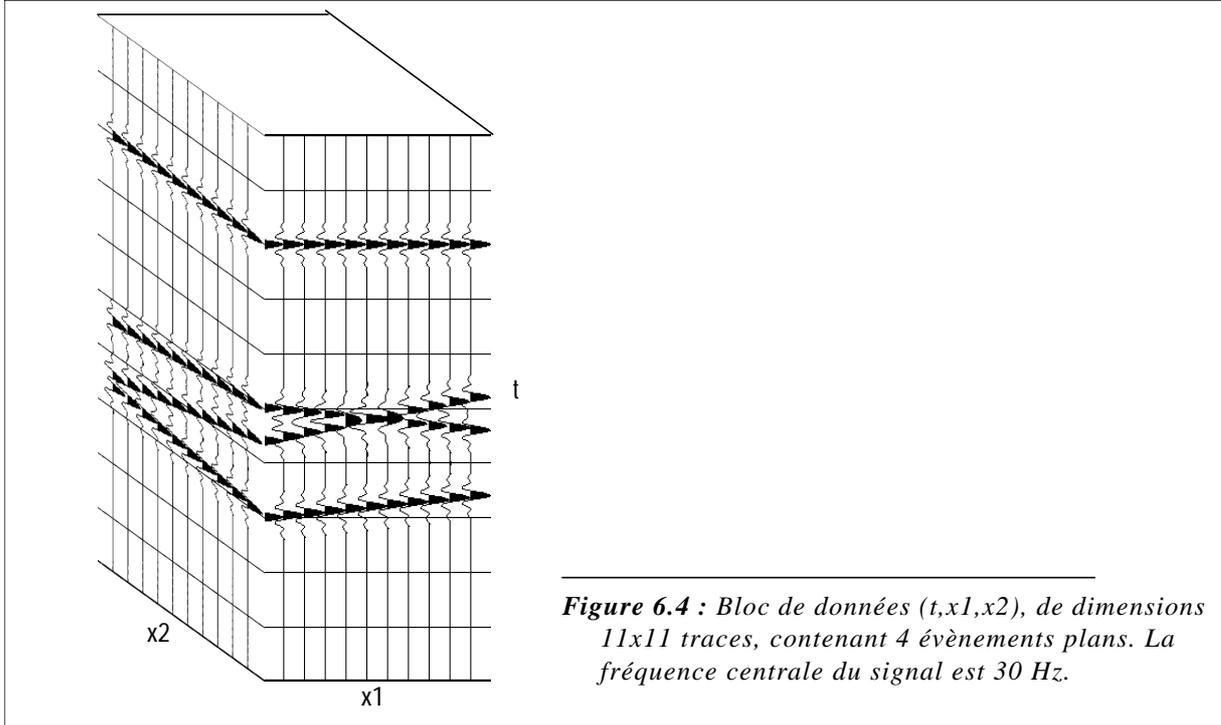


Figure 6.4 : Bloc de données $(t, x1, x2)$, de dimensions 11×11 traces, contenant 4 évènements plans. La fréquence centrale du signal est 30 Hz.

La décomposition est effectuée en prenant un modèle de 121 traces ($N_{p1}=N_{p2}=11$). Sur la figure 6.5 sont tracés pour 2 fréquences différentes (10Hz et 20Hz) le module de la matrice $\underline{\mathbf{L}}^* \underline{\mathbf{L}}$ à du système linéaire, le stack des données $\underline{\mathbf{L}}^* \underline{\mathbf{d}}$, et la solution du système $(\underline{\mathbf{L}}^* \underline{\mathbf{L}} + \varepsilon \underline{\mathbf{I}} \underline{\mathbf{d}})^{-1} \underline{\mathbf{L}}^* \underline{\mathbf{d}}$ (le facteur d'amortissement ε appliqué est 2% de la diagonale de $\underline{\mathbf{L}}^* \underline{\mathbf{L}}$).

Tous les blocs constituant la matrice $\underline{\mathbf{L}}^* \underline{\mathbf{L}}$ sont identiques à un facteur d'échelle près (traduction graphique de la factorisation donnée par les équations (E6.17) et (E6.18)). Sur le résultat de l'inversion à 20Hz, on distingue quatre taches correspondant aux quatre évènements présents dans les données (à 10Hz, l'étalement des taches est trop important pour les distinguer correctement).

Enfin, on effectue la même décomposition sur le même bloc de données mais amputé d'une trace sur deux, choisies aléatoirement. Sa géométrie n'est donc plus régulière. En conséquence, les blocs constituant la matrice $\underline{\mathbf{L}}^* \underline{\mathbf{L}}$ ne sont plus semblables entre eux (figure 6.5) (la factorisation n'est plus possible). Néanmoins, le résultat final diffère peu du cas où la géométrie est régulière (seuls les problèmes d'aliasing dus à un échantillonnage éventuellement insuffisant après avoir enlevé des traces peuvent modifier le résultat de l'inversion).

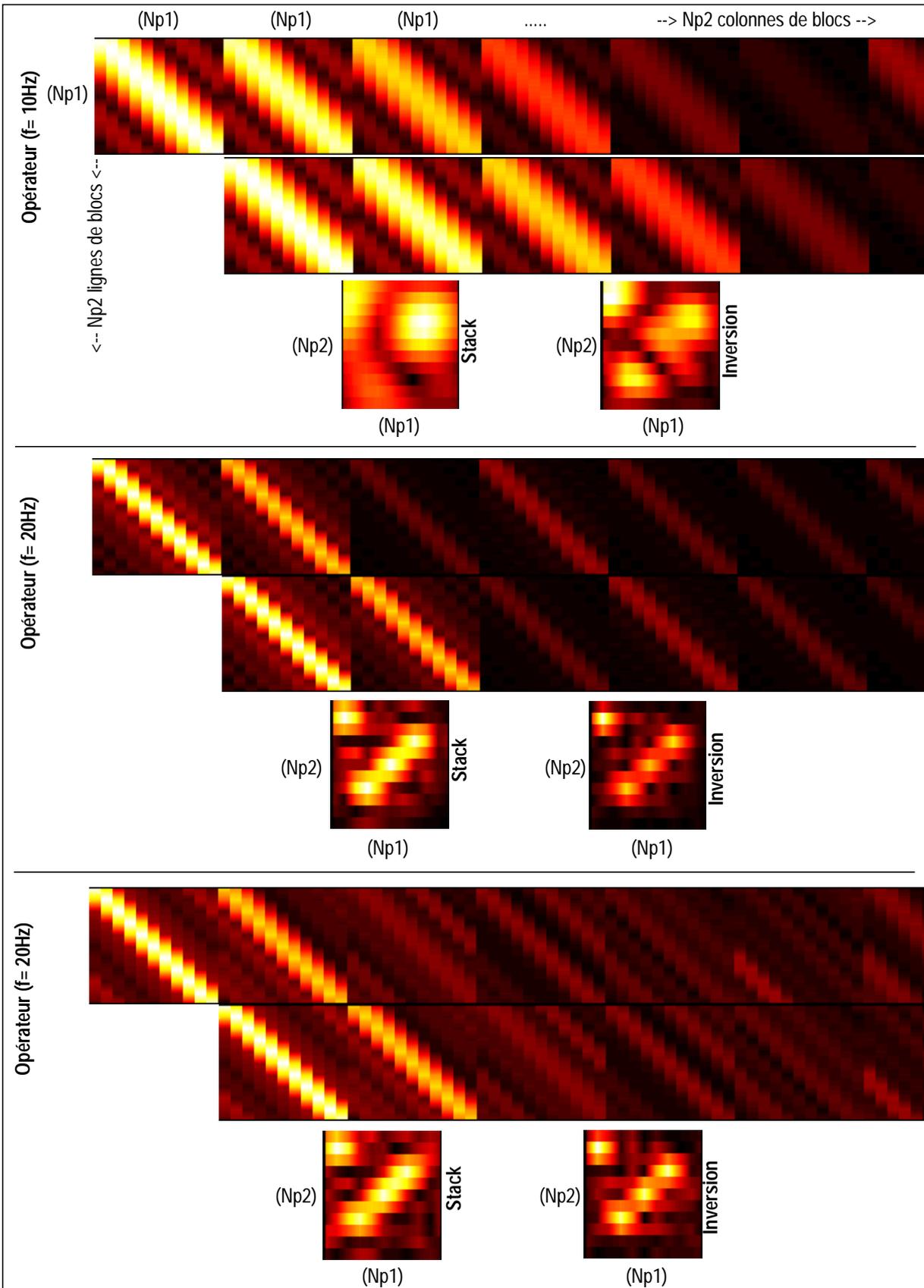


Figure 6.5 : décomposition par la méthode pour grilles irrégulières des données (régulières) fig. 6.4: modules (pour une fréquence donnée) de la matrice $\mathbf{L}^* \mathbf{L}$ (équ. (E6.13)), du stack $\mathbf{L}^* \mathbf{d}$, et de la solution $(\mathbf{L}^* \mathbf{L} + \epsilon \mathbf{Id})^{-1} \mathbf{L}^* \mathbf{d}$.
 (haut) $f=10\text{Hz}$; (milieu) $f=20\text{Hz}$; (bas) $f=20\text{Hz}$ données irrégulières (50% des traces enlevées aléatoirement).

6.4.2 - Domaine de Fourier, grille des données pseudo-régulière

Les deux matrices formant le produit dans l'équation (E6.17) peuvent s'écrire:

$$\tilde{\mathbf{Q}}_{x1} \otimes \mathbf{Id}_{Np2} = \begin{bmatrix} \tilde{\mathbf{Q}}_{x1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{Q}}_{x1} & & \\ \vdots & & \ddots & \\ \mathbf{0} & \dots & \mathbf{0} & \tilde{\mathbf{Q}}_{x1} \end{bmatrix} \quad (\text{E6.23})$$

$$\mathbf{Id}_{Np1} \otimes \tilde{\mathbf{Q}}_{x2} = \begin{bmatrix} \Sigma_{x2}(0)\mathbf{Id}_{Np1} & \Sigma_{x2}(1)*\mathbf{Id}_{Np1} & \dots & \Sigma_{x2}(Np2-1)*\mathbf{Id}_{Np1} \\ \Sigma_{x2}(1)\mathbf{Id}_{Np1} & & & \dots \\ \dots & & & \Sigma_{x2}(1)*\mathbf{Id}_{Np1} \\ \Sigma_{x2}(Np2-1)\mathbf{Id}_{Np1} & \dots & \Sigma_{x2}(1)\mathbf{Id}_{Np1} & \Sigma_{x2}(0)\mathbf{Id}_{Np1} \end{bmatrix} \quad (\text{E6.24})$$

La résolution du système global peut se décomposer en deux temps, chacun des systèmes intermédiaires étant définis par les matrices (E6.23) et (E6.24). L'ordre de résolution n'a pas d'importance (on vérifie facilement que les deux matrices commutent).

Le premier système (E6.23) correspond à $Np2$ systèmes de Toeplitz définis par la matrice $\tilde{\mathbf{Q}}_{x1}$ (dimension $Np1$), alors que le second (E6.24) correspond à $Np1$ systèmes de Toeplitz définis par la matrice $\tilde{\mathbf{Q}}_{x2}$ (dimension $Np2$). Ces résolutions requièrent donc de l'ordre de $Np2.Np1^2+Np1.Np2^2$ opérations: si par exemple $Np1=Np2$ et $Np=Np1.Np2$, alors on aboutit à un nombre d'opérations de l'ordre de $Np^{1.5}$, ce qui est une grosse amélioration par rapport aux $Np^{2.5}$ quand la géométrie des données est quelconque (paragr. précédent). Le stack réclamant quand à lui de l'ordre de Np^2 opérations, l'inversion représente cette fois une part décroissante des calculs quand la taille du système augmente.

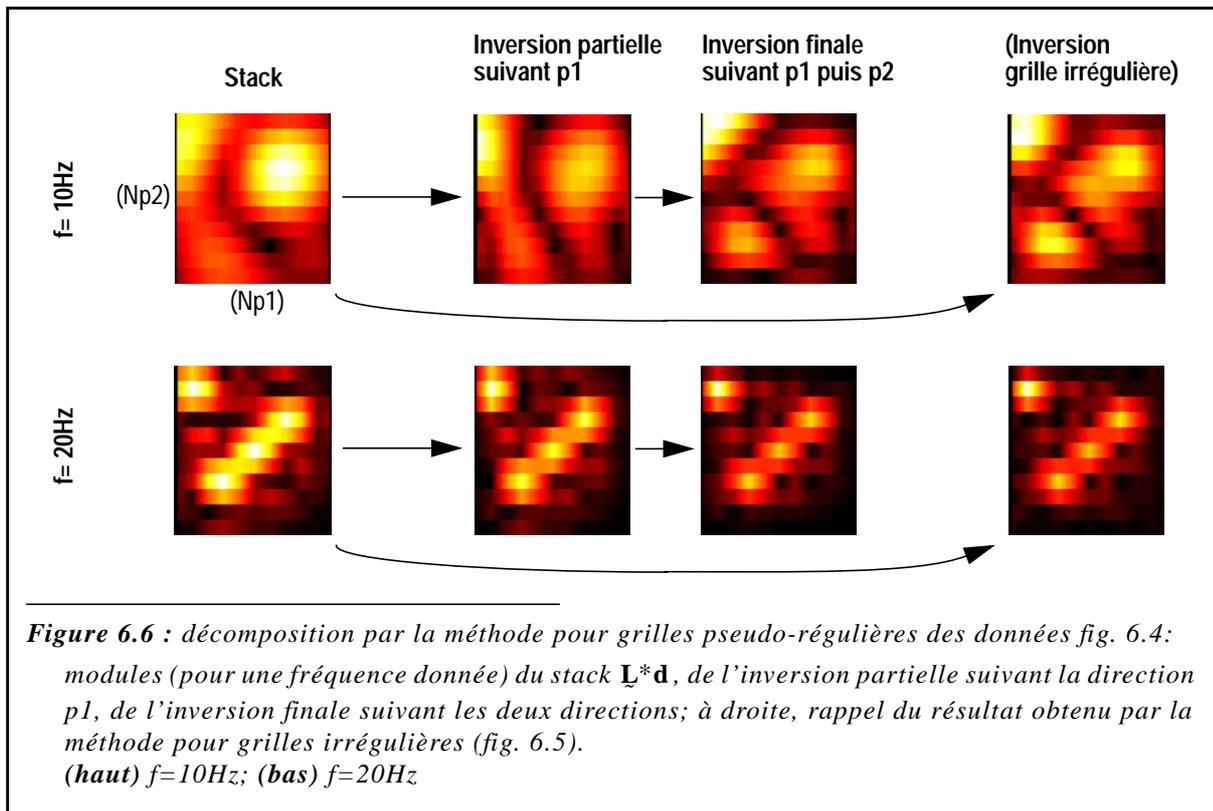
On peut reprendre les données de la figure 6.4 et effectuer leur décomposition suivant ce schéma afin de le comparer au schéma général pour géométries irrégulières (fig. 6.5). On note certaines petites différences, sans doute dues au fait que le pré-blanchiment des opérateurs s'effectue différemment dans les deux schémas: pour les géométries irrégulières, on applique un facteur d'amortissement sur la diagonale de $\underline{\mathbf{L}}*\underline{\mathbf{L}}$, alors que dans le présent schéma, on applique le facteur d'amortissement (choisi égal à 2% dans cet exemple) sur les diagonales des matrices $\tilde{\mathbf{Q}}_{x1}$ et $\tilde{\mathbf{Q}}_{x2}$. Cela revient à appliquer un facteur d'amortissement à la diagonale de chacun des blocs constituant $\underline{\mathbf{L}}*\underline{\mathbf{L}}$, auquel se superpose un facteur d'amortissement appliqué à sa diagonale principale de blocs (voir équ. (E6.17) et (E6.18)).

6.4.3 - Domaine temporel, tous types de grilles

Le domaine temporel offre l'avantage notable par rapport au domaine de Fourier de pouvoir traiter indifféremment, avec le même algorithme, tous les types de grilles, non seulement pour les données mais également pour le modèle (en Fourier, on est limité à une grille régulière rectangulaire pour le modèle).

On utilise exactement le même schéma que pour les décompositions 2D, à savoir un gradient conjugué.

Nous avons vu (équ. (E6.6)) que la LRD 3D, pour des variables continues, pouvait s'écrire à l'aide des opérateurs de la LRD 2D, l'opérateur $\underline{\mathbf{L}}*\underline{\mathbf{L}}$ étant la convolution successive dans chacune des



directions p1 et p2 par les opérateurs 2D. Les effets fréquentiels de ces opérateurs (voir parag. 3.8.2) se superposent donc, et leurs conséquences sur la convergence peuvent être très importantes, même pour des données d'extension spatiale limitée.

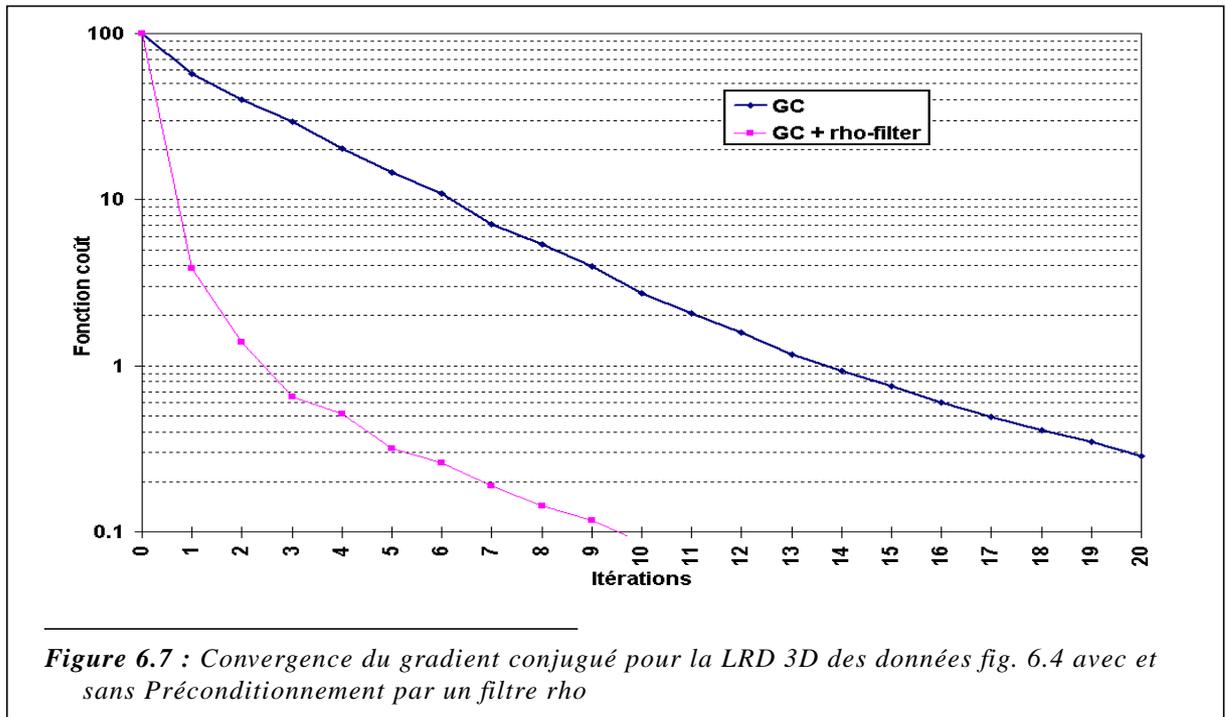
Nous préconditionnons donc le gradient conjugué par un filtre rho construit par la multiplication des deux filtres rho 2D (définis par l'équ. (E3.53)) pour chacune des directions p1 et p2. Un des problèmes qui se pose est de définir les fréquences limites d'aliasing spatial pour ces directions lorsque la grille des données est irrégulière: dans une telle grille, comment définir les pas d'échantillonnages spatiaux Δx_1 et Δx_2 , dont dépendent ces fréquences d'aliasing? Une solution est de choisir (plus ou moins arbitrairement?) ces pas d'échantillonnage, en fonction par exemple des valeurs nominales de l'acquisition.

L'amélioration apportée par le filtre rho sur la convergence du GC est spectaculaire (fig. 6.7, pour la LRD 3D des données fig. 6.4).

Un filtre rho approprié permet donc de reconditionner le problème par rapport aux effets fréquentiels. L'autre cause importante affectant la convergence est l'irrégularité de la grille des données: si on considère un événement plan d'amplitude constante mais d'extension spatiale limitée, les amplitudes concernant cet événement après application de l'opérateur $\mathbf{L}^* \mathbf{L}$ vont être directement proportionnelles au nombre de traces sur lequel il est représenté dans les données (par l'opération de stack \mathbf{L}^* on somme autant de fois l'amplitude de l'événement que le nombre de trace sur lequel il est représenté). La répartition spatiale des traces influence donc directement le nombre de conditionnement du système, dans une proportion approximativement égale au rapport des densités minimales et maximales de traces dans les données (Feichtinger, 1995).

On peut reprendre l'exemple de la figure 6.4, en considérant un bloc plus grand de 21x21 traces, dans lequel on enlève d'une part une ligne sur deux et une trace sur deux dans chaque ligne, pour arriver à un bloc régulier de 11x11 traces, et d'autre part des traces au hasard de manière à obtenir un bloc irrégulier de 121 traces (fig. 6.8). L'irrégularité ralentit notablement la convergence pendant les premières itérations.

Une façon de compenser l'irrégularité est d'appliquer une pondération sur les données: au lieu de minimiser la quantité (équ. 3.36):



$$\varphi(\mathbf{u}) = \frac{1}{2} \|\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}\|^2 + \frac{\varepsilon}{2} \|\mathbf{u}\|^2,$$

on minimise plutôt:

$$\varphi(\mathbf{u}) = \frac{1}{2} (\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u}, \mathbf{P}_{\text{irr}}(\mathbf{d} - \tilde{\mathbf{L}}\mathbf{u})) + \frac{\varepsilon}{2} \|\mathbf{u}\|^2 \quad (\text{E6.25})$$

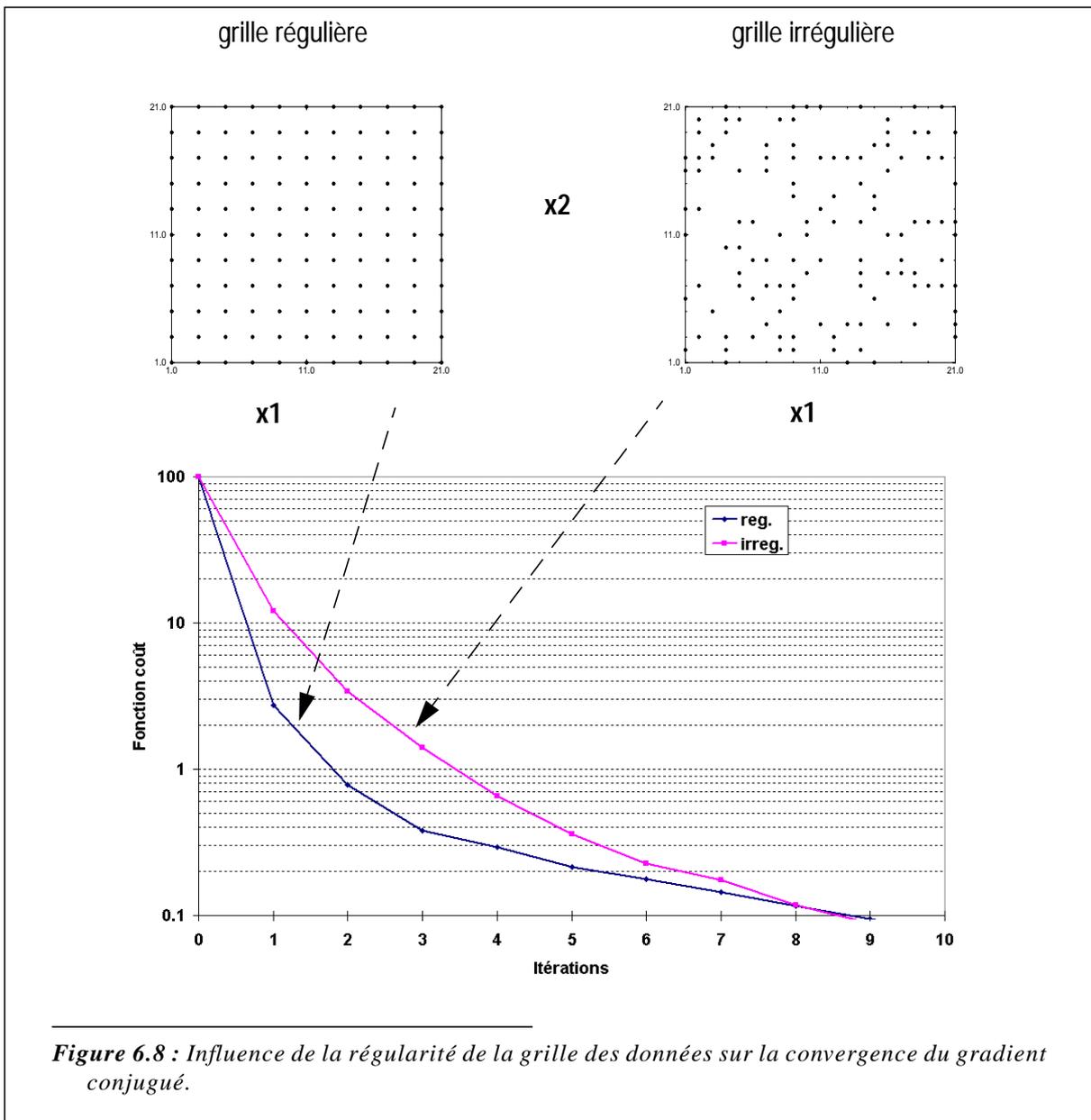
où \mathbf{P}_{irr} est un poids diagonal qui contient par exemple pour chaque trace la distance entre les deux traces encadrant celle-ci (Feichtinger, 1995; Duijndam, 1995) pour un problème 2D... En 3D cela peut être la 'surface associée' à chaque trace, qui contient tous les points pour lesquels cette trace est la plus proche). Ce poids doit être faible dans les zones à forte densité de traces et réciproquement.

S'il est vrai qu'un tel poids reconditionne le problème et accélère la convergence, sa justification n'est pas évidente (problème d'ailleurs évoqué par Duijndam, 1995), car on change non seulement la façon de résoudre le problème, mais aussi le problème lui-même. Suivant le point de vue probabiliste de l'inversion, le poids est l'inverse de la matrice de covariance décrivant le bruit (supposé gaussien) présent sur les données (équ. (E3.2)). Or des traces très proches entraînent un poids faible, donc une variance élevée, signifiant que l'on accorde peu de confiance à la mesure, ce qui n'est pas justifié (dans un problème sans bruit, ces considérations n'auraient bien sûr pas lieu d'être).

La vraie solution est un peu plus complexe. Considérons deux traces et un échantillon au même temps sur chaque trace. Si les deux traces sont suffisamment éloignées, la matrice de covariance sur les données pour ces deux échantillons est diagonale, traduisant que les incertitudes (le bruit 'aléatoire' + le bruit de

modélisation) sont décorrélées (supposons également qu'elles sont égales): $\mathbf{C}_{\mathbf{d}} = \begin{bmatrix} c_{11} & 0 \\ 0 & c_{11} \end{bmatrix}$. Le poids

correspondant est $\mathbf{P}_{\text{irr}} = \begin{bmatrix} c_{11} & 0 \\ 0 & c_{11} \end{bmatrix}^{-1} = \begin{bmatrix} 1/c_{11} & 0 \\ 0 & 1/c_{11} \end{bmatrix}$



Si maintenant on rapproche les traces, il apparaît un coefficient de covariance c_{12} : $\tilde{\mathbf{C}}_d = \begin{bmatrix} c_{11} & c_{12} \\ c_{12} & c_{11} \end{bmatrix}$,

traduisant une corrélation des incertitudes. Si il n'y a aucun bruit de modélisation ce coefficient est nul. Si il n'y a pas de bruit aléatoire et que les traces sont situées au même point, il est égal à c_{11} .

Le poids à appliquer doit être l'inverse de cette matrice:

$$\tilde{\mathbf{P}}_{irr} = \begin{bmatrix} c_{11} & c_{12} \\ c_{12} & c_{11} \end{bmatrix}^{-1} = \frac{1}{c_{11}^2 - c_{12}^2} \begin{bmatrix} c_{11} & -c_{12} \\ -c_{12} & c_{11} \end{bmatrix} = \frac{1}{1 - (c_{12}/c_{11})^2} \begin{bmatrix} 1/c_{11} & -c_{12}/c_{11} \\ -c_{12}/c_{11} & 1/c_{11} \end{bmatrix}$$

On constate effectivement que le poids diminue sur la diagonale, mais aussi qu'il apparaît des éléments non diagonaux: l'un ne va pas sans l'autre.

Dans la pratique, il est complètement irréaliste d'appliquer des poids non diagonaux dans des problèmes à plusieurs dizaines de milliers d'inconnues. D'un autre côté, ne retenir que la diagonale n'est pas plus justifié que de ne pas appliquer de poids...

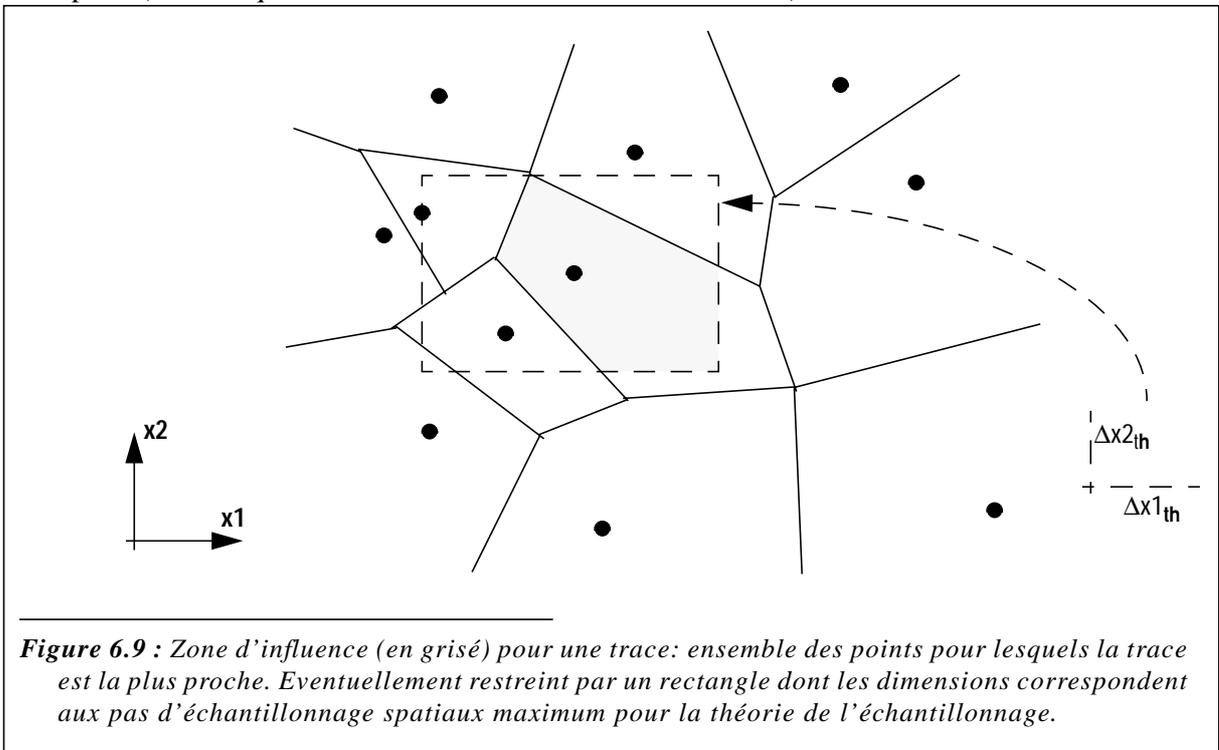
On peut néanmoins introduire ce poids diagonal décrit par Duijndam, non pas dans la fonction coût, mais seulement pour modifier le calcul du gradient.

Pour la minimisation de $\varphi(\mathbf{u}) = \frac{1}{2}\|\mathbf{d} - \mathbf{L}\mathbf{u}\|^2 + \frac{\varepsilon}{2}\|\mathbf{u}\|^2$, le gradient vaut: $\mathbf{g}(\mathbf{u}) = -\mathbf{L}^*(\mathbf{d} - \mathbf{L}\mathbf{u})$. On veut ici utiliser un gradient 'modifié':

$$\mathbf{g}_{\text{irr}}(\mathbf{u}) = -\mathbf{L}^* \mathbf{P}_{\text{irr}}(\mathbf{d} - \mathbf{L}\mathbf{u}) \quad (\text{E6.26})$$

Comme il ne s'agit pas du 'vrai' gradient pour le problème quadratique posé, les formules de conjugaison ne sont pas applicables, et à chaque itération il faut effectuer une recherche bilinéaire (avec ce gradient modifié et la direction de descente précédente), qui est heureusement négligeable en termes de temps de calcul.

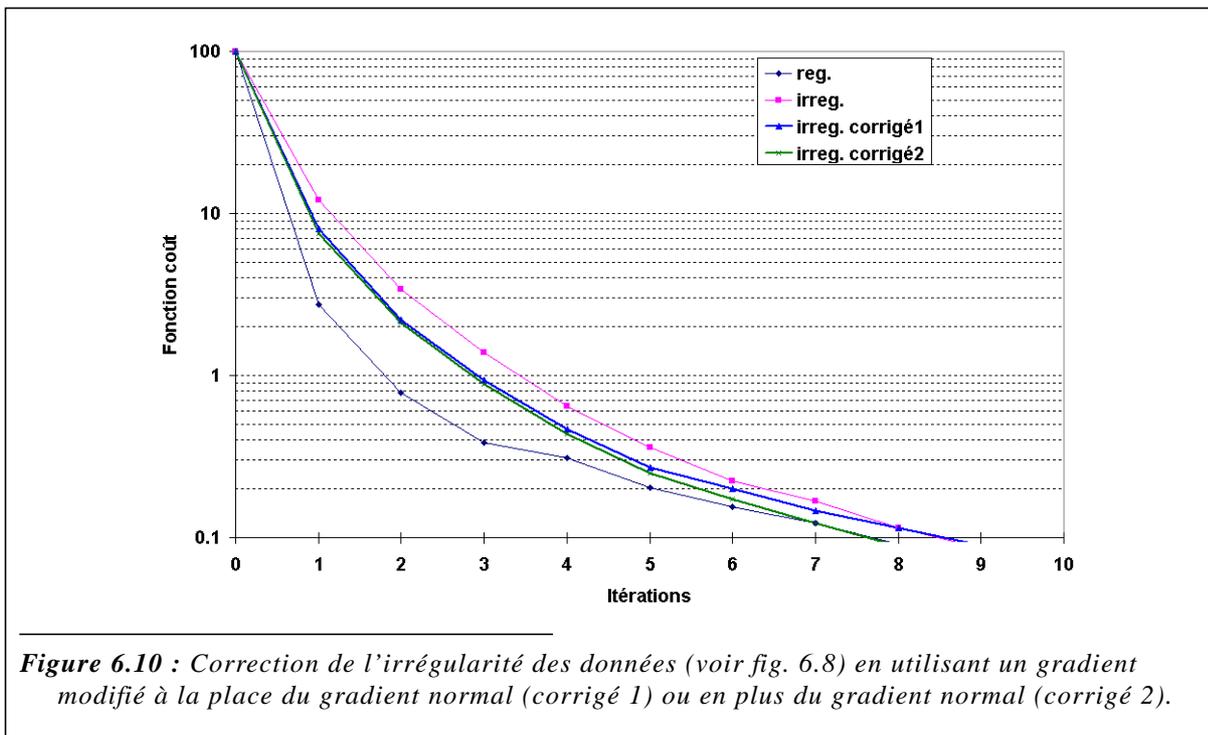
Le poids est calculé d'après la surface de la zone d'influence de la trace (fig. 6.9), déterminée par l'ensemble des points pour lesquels cette trace est la plus proche. On limite cette zone qui doit s'inscrire dans un rectangle centré sur la trace dont les dimensions sont les pas d'échantillonnages spatiaux maximum $\Delta x_{1\text{th}}$, $\Delta x_{2\text{th}}$ donnés par la théorie de l'échantillonnage (équ. (E2.5)). On considère en effet qu'un point situé au-delà de ces valeurs n'est plus influencé par la trace, et qu'il y a absence de données en ce point (à moins qu'il ne soit sous l'influence d'une autre trace).



Ce type de pondération est utilisé dans d'autres problèmes où il est bénéfique de corriger l'irrégularité des données (voir par exemple Beasley and Klotz, 1992, pour le DMO)

Avec ce gradient modifié appliqué à l'exemple précédent (fig. 6.8), la convergence est plus rapide, principalement sur les premières itérations (fig. 6.10), sans pour autant atteindre la même performance que sur la grille régulière. Néanmoins l'irrégularité n'est ici pas très sévère, et on peut supposer que dans les cas plus sévères ce poids peut compenser la majeure partie de l'irrégularité.

En n'utilisant pas le vrai gradient dans le processus de gradient conjugué, on n'est pas assuré théoriquement de pouvoir atteindre le minimum. On essaie donc d'utiliser le gradient modifié non pas *à la place* mais *en plus* du gradient vrai: à chaque itération on effectue une recherche trilineaire avec le gradient vrai, le gradient modifié, et la direction de descente précédente. On a également l'espoir d'accélérer encore la convergence en utilisant une direction de recherche supplémentaire. Or il semble (fig. 6.10) que cela ne soit le cas qu'à partir du moment où on est déjà proche du minimum. Dans les premières itérations, les plus importantes, l'utilisation du gradient vrai en plus du gradient modifié n'apporte aucune amélioration: les deux courbes sont pratiquement confondues.



Au niveau temps de calcul, l'utilisation de deux 'gradients' est très pénalisante, car dans chaque itération il faut appliquer les opérateurs \underline{L}^* et \underline{L} pour chacun d'eux, ce qui conduit pratiquement à doubler le temps de calcul. L'utilisation du gradient modifié seul à la place du gradient vrai ne requiert par contre que très peu de calculs supplémentaires (l'application de \underline{P}_{irr} , qui est diagonal, sur le résidu $(\underline{d} - \underline{L}\underline{u})$). Le peu d'amélioration apportée par l'utilisation conjointe des deux gradients est donc balayée par le temps de calcul supplémentaire. Dans la pratique, on n'utilise que la première implémentation avec le gradient modifié seul.

Le filtre rho permet d'obtenir une très bonne approximation de la solution avec 1 itération seulement pour une grille des données régulière. Pour des grilles irrégulières, l'utilisation du gradient modifié permet d'obtenir une bonne convergence en 5 itérations dans semble-t-il tous les cas de figure. Il est important de pouvoir fixer un nombre d'itérations à priori pour tous les problèmes traités: le nombre d'opérations nécessaires pour une décomposition ne dépend alors plus que des opérateurs \underline{L}^* et \underline{L} , appliqués chacun une fois dans chaque itération et qui représentent la plus grosse part des calculs. Ils réclament de l'ordre de $N_x \cdot N_p \cdot N_t$ opérations, soit de l'ordre de $N_p^2 \cdot N_t$ dans une configuration courante où $N_p = N_x$.

6.4.4 - Performances comparées des différentes implémentations

On compare ici les performances des 2 implémentations, en domaine temps par gradient conjugué (GC), en domaine de Fourier par des méthodes directes (Four.). Pour des données sur grille régulière, l'algorithme de résolution en Fourier (Four. Reg.) doit être comparé avec une seule itération de gradient conjugué (GC 1iter) qui dans ce cas est suffisante pour aboutir à la solution. Pour des données sur grille irrégulière, l'algorithme de résolution en Fourier (Four. Irr.) doit être comparé avec 5 itérations de gradient conjugué (GC 5iter). Les quatre jeux de données consistent en des blocs de 5x5, 10x10, 20x20, et 40x40 traces de 256 échantillons temps. Le nombre de traces utilisées dans les modèles est à chaque

fois égal au nombres de traces des blocs de données. On effectue le test sur un Sun UltraSparc (architecture super-scalaire, Tableau 2 :), et sur un Cray C92 (architecture vectorielle, Tableau 3 :).

Tableau 2 : Temps CPU (sec.) sur Sun UltraSparc (super-scalaire) pour la LRD 3D en norme L2 de données de tailles différentes

	5x5	10x10	20x20	40x40
GC 1iter	0.23	1.09	9.46	52.50
Four. Reg.	0.38	3.37	41.80	644.00
GC 5iter	0.75	4.60	42.90	245.00
Four Irr.	1.11	16.50	370.00	10800.00

Tableau 3 : Temps CPU (sec.) sur Cray C92 (vectoriel) pour la LRD 3D en norme L2 de données de tailles différentes

	5x5	10x10	20x20	40x40
GC 1iter	0.04	0.17	1.18	11.00
Four. Reg.	0.09	0.42	3.60	46.40
GC 5iter	0.12	0.57	4.68	42.90
Four Irr.	0.53	4.61	56.90	934.00

Dans tous les cas, l'implémentation en domaine temps par gradient conjugué est plus efficace que l'implémentation en domaine de Fourier. De plus, l'écart s'accroît en faveur du domaine temps quand la taille du problème augmente, surtout pour les grilles irrégulières où il devient énorme.

Au vu de ces résultats nous n'utilisons dans la pratique que le gradient conjugué en domaine temps, qui offre par ailleurs de nombreux avantages (traitement correct des mutes, application de poids variables en temps,...). Pour être juste, il faut néanmoins signaler que les implémentations en domaine de Fourier sont beaucoup moins gourmande en mémoire, chaque fréquence pouvant être traitée indépendamment des autres.

6.5 - Décompositions parcimonieuses

Tous les principes des décompositions parcimonieuses défini pour les décompositions 2D dans le chapitre 4 sont directement applicables à la LRD 3D.

Tout ce que l'on peut dire de plus pour le cas 3D, c'est que si l'hypothèse de parcimonie est justifiée en 2D elle l'est encore plus en 3D. En effet entre un bloc 3D et une section 2D de ce bloc, on retrouve en principe le même nombre d'évènements à décomposer. La densité des évènements dans les modèles est donc en principe plus faible dans un espace 3D que dans un espace 2D. Toute dimension supplémentaire est en principe bénéfique du point de vue de la parcimonie.

6.6 - Conclusions

La LRD 3D est une extension triviale de la LRD 2D au niveau du formalisme.

Les implémentations en domaine temps par gradient conjugué, que ce soit en norme L2 ou pour les décompositions parcimonieuses, ne diffèrent pas non plus de la LRD 2D. Le filtre rho utilisé, indispensable pour obtenir une bonne convergence, est construit à partir des filtres rho 2D. Si la géométrie des données est irrégulière, il est utile de corriger le gradient calculé par un poids contenant la surface relative autour de chaque trace.

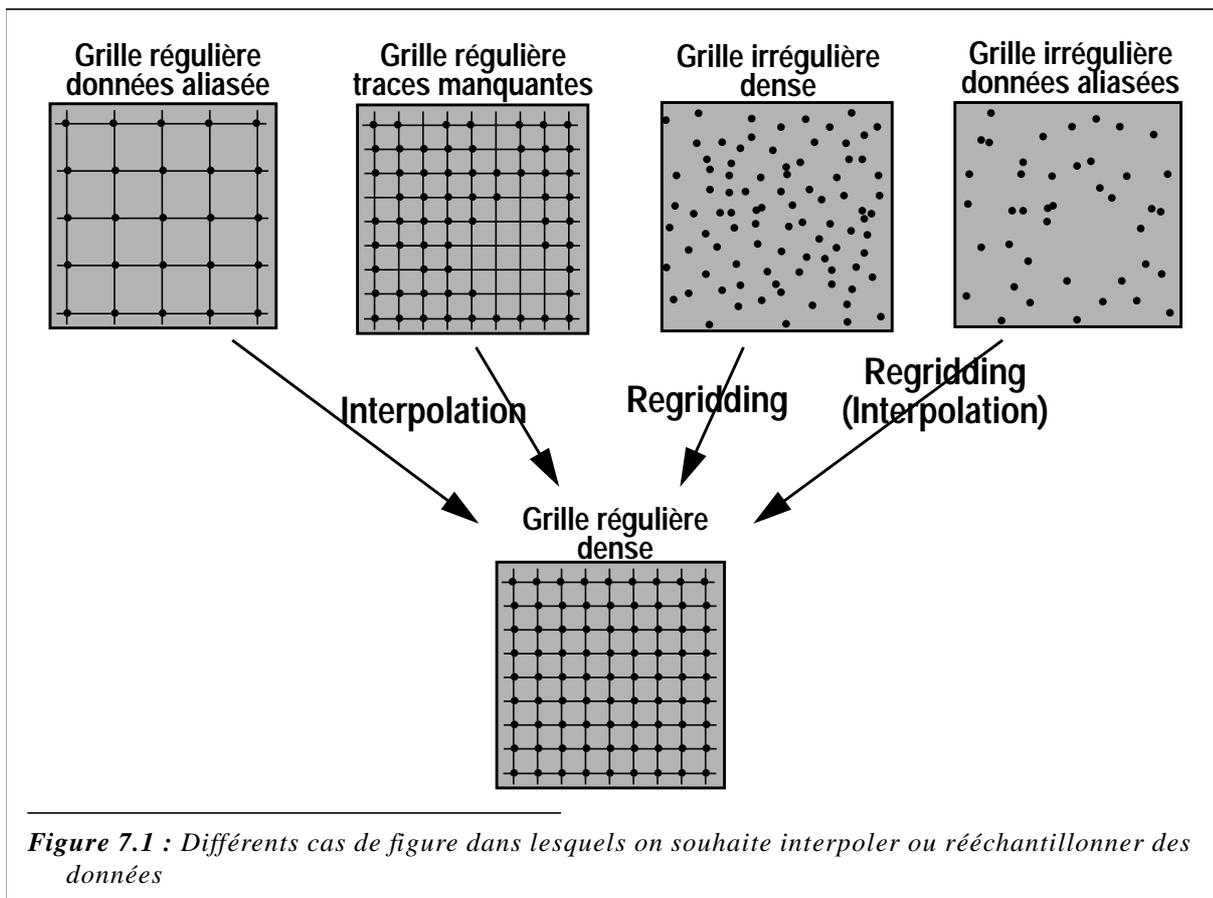
Dans le domaine de Fourier, pour la résolution en moindres carrés, on obtient une résolution efficace par des méthodes directes uniquement si les données sont sur une grille pseudo-régulière rectangulaire, auquel cas le problème 3D se décompose en 2 fois 2D. Sinon on aboutit à la résolution d'un système Toeplitz par blocs, pour lequel il n'existe pas de méthode réellement efficace.

*7 - Rééchantillonnage et
Interpolation de traces
(2D/3D) par les
décompositions de Radon*

7.1 - Introduction

L'interpolation de traces est un problème récurrent en sismique. On peut distinguer quatre situations dans lesquelles on peut vouloir interpoler (voir fig. 7.1):

- (a) Les données sont sur une grille régulière, mais sont spatialement aliasées. On veut donc réduire les pas d'échantillonnages spatiaux et pour cela interpoler de nouvelles traces.
- (b) Les données sont sur une grille régulière suffisamment dense, mais sur laquelle il y a des traces manquantes que l'on veut reconstruire.
- (c) Les données sont sur une grille quelconque (généralement irrégulière) suffisamment dense pour qu'il n'y ait pas d'aliasing spatial. On veut les rééchantillonner sur une autre grille (regridding) (généralement régulière).
- (d) Les données sont sur une grille quelconque, et sont spatialement aliasées. On veut les rééchantillonner sur une autre grille.



Il n'y a que dans le cas (c) que toute l'information nécessaire pour interpoler sans ambiguïté est contenue dans les données (Marvasti et al., 1991; Feichtinger et al., 1995; Duinjdram et al., 1995): la supposition de non-aliasing revenant à dire que les données sont à bandes de fréquences spatiales limitées et que l'échantillonnage est suffisant par rapport à celles-ci. Dans les autres cas les données sont spatialement aliasées, ce qui signifie qu'il faut d'une manière ou d'une autre introduire certains a priori pour contraindre l'interpolation (le cas (b) peut éventuellement être assimilé au cas (c) si la grille est sur-échantillonnée -situation peu probable sur des données réelles-, auquel cas la redondance d'information autorise la perte de certaines traces).

Le cas (c) est néanmoins ambigu car si les données sont disposées sur une grille très irrégulière, la condition de non-aliasing peut être respectée *en moyenne*, mais pas *localement*. Il faut pour cela que l'échantillonnage soit suffisamment dense en tout point de l'espace. Cela peut paraître évident mais une grille très irrégulière peut conduire à un sur-échantillonnage à certains endroits et un sous-échantillonnage à d'autres, une valeur moyenne n'a donc aucune signification¹.

Les cas (a) et (b) sont fondamentalement très différents des cas (c) et (d). Dans les premiers, on ajoute de nouvelles traces interpolées à l'ensemble des données de départ -qui sont conservées-, d'après le principe qu'il vaut mieux avoir une donnée approximative en tout noeud de la grille qu'une absence de données. On ne cherche donc pas une interpolation parfaite à tout prix. Dans les deux autres cas, l'ensemble des données de départ est remplacé par les données interpolées. L'interpolation doit donc être de haute qualité car on ne peut pas se permettre de perdre de l'information.

Le principe selon lequel mieux vaut mieux une donnée approximative qu'une absence de données est en toute rigueur une contre-vérité. On sait bien que l'interpolation ne crée pas d'information, tout au plus crée-t-elle des traces qui sont cohérentes avec les traces de départ (suivant certains critères à définir). Néanmoins, de nombreux traitements sont effectués en supposant que les données sont disponibles en tout point de l'espace (donc non aliasées) et distribuées sur des grilles régulières (ne serait-ce que le stack, mais aussi les implémentations les plus courantes du DMO, les migrations,...). Tout écart à ces hypothèses se traduit par une perte de qualité plus ou moins importante, d'où le besoin d'interpolation pour avoir les données échantillonnées de manière optimum. La raison sous-jacente est que la plupart des traitements sismiques sont des opérateurs transposés utilisés comme approximations des véritables opérateurs inverses (Claerbout, 1992): une migration de Kirchhoff est la transposée d'une modélisation. Cette approximation est en général suffisante si les données sont non-aliasées et distribuées sur des grilles régulières. Si ce n'est pas le cas, il faut concevoir les traitements différemment, suivant des techniques d'inversion, ce qui complique la conception des algorithmes (Inversion to Zero Offset à la place du classique DMO de Kirchhoff, Ronen et al., 1995) et augmente le temps de calcul.

Les méthodes d'interpolation peuvent être classées en deux grandes catégories: les méthodes uniquement basées sur un modèle indépendant des données, et les méthodes pilotées par les données. Dans première catégorie, on définit un modèle dans lequel sont censées s'inscrire les données: par exemple qu'elles sont à bandes de fréquences limitées (Duinjdram et al., 1995), ou qu'elles obéissent à l'équation de propagation des ondes dans un milieu connu (Ronen, 1987). Les valeurs prises sur les positions interpolées ne sont alors pas ambiguës mais imposées par le modèle et non ambiguës. Dans la deuxième catégorie, le modèle n'est pas assez contraignant pour interpoler sans ambiguïtés, et il faut déduire des données elles-mêmes comment interpoler, suivant des critères à définir. Les méthodes basées sur des filtres prédictifs (Spitz, 1991; Huard and Spitz, 1996) appartiennent à cette catégorie. Les filtres sont calculés à partir des données disponibles, et sont ensuite utilisés pour reconstituer les données manquantes.

L'interpolation de traces est une application assez courante des décompositions de Radon. On utilise généralement les décompositions de Radon linéaires (Lu, 1991; Stoffa et al., 1981), permettant d'interpoler des événements de toutes formes. On peut également utiliser les Décompositions Hyperboliques ou Paraboliques sur les collections CMP avant ou après NMO (Darche, 1990; Sacchi and Ulrych, 1995).

¹. *En principe, un signal périodique à bande limitée peut être déterminé de manière unique à partir d'un nombre fini de points d'échantillonnage, indépendamment de la répartition de ces derniers. La régularité de l'échantillonnage permet simplement de stabiliser les algorithmes de reconstruction (voir par exemple Marvasti et al., 1991). En théorie, on devrait donc pouvoir interpoler à travers de larges gaps, pour peu que l'échantillonnage soit suffisamment dense 'en moyenne'. Il n'est en fait pas possible de tenir ce raisonnement dans la pratique, car les données réelles ne peuvent pas être considérées strictement à bande limitée: ainsi toute discontinuité (due à la géologie par exemple) introduit des hautes fréquences. En conséquence, un échantillonnage très serré dans une zone n'apporte pas forcément les informations pour compenser un sous-échantillonnage dans une autre zone. De toutes façons, même dans le cas théorique idéal de bande limitée, le problème est en général trop mal conditionné pour être résolu de manière efficace s'il y a des gaps dans les données.*

Les décompositions en norme L2 rentrent dans la première catégorie d'interpolateurs (le modèle consiste à expliquer les données par une superposition de droites, de plans, de paraboles...). Elles ne traitent pas correctement les données spatialement aliasées, de même qu'une transformée de Fourier ne permet pas d'interpoler un signal aliasé (une décomposition de Radon en norme L2 est d'ailleurs pratiquement équivalente à un spectre (f,k))

Le problème d'aliasing spatial est souvent résolu en pondérant les sommations le long des courbes de modélisation par des semblances locales afin de privilégier la continuité des événements cohérents (Lu, 1991). Ces méthodes ne sont toutefois pas des inversions, mais restent dans le domaine de la transformée de Radon améliorée: ainsi le cas d'évènements interférents est généralement mal résolu, seul le plus énergétique étant correctement interpolé. Yilmaz and Taner (1994) ont construit une véritable décomposition par inversion à partir de ce principe de semblance, mais outre que leur méthode semble coûteuse, elle a le désavantage d'utiliser un opérateur de modélisation \mathbf{L} différent des opérateurs de Radon habituels, et surtout non linéaire car dépendant des données elles-mêmes. L'utilisation des décompositions parcimonieuses se situe dans la continuité des méthodes de semblances, mais avec tous les avantages d'une décomposition rigoureuse (tous les événements sont en principe correctement interpolés) (Sacchi and Ulrych, 1995). Les décompositions parcimonieuses rentrent dans la seconde catégorie d'interpolateurs, puisque l'on favorise les événements spatialement cohérents dans le processus. L'hypothèse sous-jacente (l'apriori introduit dans le processus d'inversion) est que les données contiennent un nombre limité d'évènements superposés et indépendants: c'est la même que celle utilisée dans les méthodes à base de filtres prédictifs. Cette hypothèse exclut les phénomènes dispersifs (où la vitesse apparente, donc la pente p , varie avec la fréquence).

Un des principaux intérêts des décompositions de Radon est de pouvoir traiter indifféremment les grilles régulières et irrégulières. Les méthodes basées sur des filtres prédictifs n'ont pas cette souplesse, car bien qu'il soit possible de traiter le cas des grilles irrégulières, les algorithmes sont beaucoup plus complexes et instables (on passe d'un problème linéaire à un problème fortement non-linéaire, voir Claerbout, 1992). Le problème de la reconstruction d'un signal (mono ou multi-dimensionnel) à partir d'un échantillonnage irrégulier a été abondamment traité d'un point de vue théorique ces dernières années (voir par exemple Marvasti, 1991), l'attention se portant maintenant sur la mise au point d'algorithmes efficaces (Feichtinger, 1995). De ce point de vue, l'utilisation de pondérations destinées à tenir compte de l'irrégularité de l'échantillonnage suffit à obtenir pratiquement les mêmes performances sur les grilles irrégulières que sur les grilles régulières. Enfin on peut obtenir un temps de calcul strictement proportionnel à la quantité de données à traiter en découpant le domaine des données en vignettes de tailles prédéfinies, vignettes que l'on traite ensuite indépendamment les unes des autres pour interpoler.

7.2 - Distances d'interpolation/extrapolation maximum en norme L2 (exemple sur LRD 2D)

Nous cherchons ici à vérifier expérimentalement la distance maximum d'interpolation ou d'extrapolation permise par des décompositions de Radon en norme L2. En théorie celle-ci doit être de l'ordre du pas d'échantillonnage spatial nécessaire. Pour la LRD 2D par exemple, ce dernier est donné par la relation (E2.12):

$$\Delta x \leq \frac{1}{f|p - p^*|_{\max}} \quad (E7.1)$$

où p^* est la pente d'un événement particulier. Ce pas dépendant de la fréquence considérée, des données monofréquentielles (fig. 7.2a) sont plus pratiques pour interpréter les tests effectués. Comme les décompositions en norme L2 sont obtenues par des opérateurs linéaires (équ. (E3.37)), il est tout à fait licite d'étudier un signal monofréquentiel et d'extrapoler le résultat à un signal quelconque.

Les données (fig. 7.2a) sont constituées de 21 traces d'abscisses comprises entre 0 et 1000m (intertrace constante $\Delta x=50m$), et contiennent un événement horizontal ($p^*=0$) portant un signal

monofréquentiel à $f=50$ Hz. La gamme de pentes maximum étudiable sans aliasing est donc:
$$p_{\text{range}} = \frac{1}{f\Delta x} = \frac{1}{2500} = 0.0004 \text{ s/m}$$
. L'évènement n'est donc pas aliasé si on se restreint à une gamme $-0.0004 < p < 0.0004 \text{ s/m}$. Toutes les LRD sont ici effectuées avec une gamme symétrique $[-p_{\text{max}}, p_{\text{max}}]$.

Le premier test consiste à interpoler et extrapoler les données en reconstruisant, à partir des LRD, les traces entre les abscisses -100m et 1100m, au pas de 12.5m (4 fois plus de traces que les données). On calcule deux LRD avec $p_{\text{max}}=0.0003 \text{ s/m}$ (fig. 7.2b) et avec $p_{\text{max}}=0.0005 \text{ s/m}$ (fig. 7.2d), c'est à dire au-dessous et au-dessus de la limite d'aliasing. Les reconstruction correspondantes sont en fig. 7.2c/e. Dans le premier cas l'interpolation est quasi-parfaite, dans le deuxième elle est complètement fautive: ni les amplitudes ni les phases des traces interpolées ne correspondent à l'évènement horizontal d'origine. Il suffit donc d'être juste au delà de la condition d'aliasing pour ne plus pouvoir interpoler du tout. En ce qui concerne l'extrapolation: dans le premier cas les traces à -12.5m et 1012.5m sont correctes, dans le second cas aucune des traces n'est correctement extrapolée (les pas d'échantillonnage limites correspondant aux pentes utilisées sont respectivement 66m et 40m).

Le second test consiste à interpoler à travers un gap de largeur variable (100m, 150m, et 200m, fig. 7.3abc). Les LRD sont toutes effectuées avec $p_{\text{max}}=0.00015 \text{ s/m}$, ce qui correspond à un pas d'échantillonnage limite égal à 133m. On constate à nouveau que dès que le gap dépasse le pas limite, l'interpolation, parfaite à travers le gap de 100m (fig. 7.3a), est dégradée (fig. 7.3bc). La distance d'extrapolation est ici de l'ordre de 25m, toujours très faible comparée au pas d'échantillonnage limite.

En conclusion le pas d'échantillonnage limite donné par les relations (E2.5) ou (E7.1) est une limite assez stricte pour l'interpolation. La portée de l'extrapolation est quant à elle encore plus réduite par rapport à cette valeur.

7.3 - Utilisation des propriétés anti-aliasing des décompositions parcimonieuses (exemple sur PRD 2D)

Considérons le jeu de données synthétique de la figure 4.2(a). Il s'agit d'une collection CMP corrigée de NMO. A partir de ces données amputées d'une trace sur deux, on veut reconstruire les traces manquantes. On utilise pour cela des décompositions paraboliques. Dans un premier temps, les réflecteurs sont habillés d'un ricker d'ordre 2 de fréquence centrale $f_c=40\text{Hz}$. La gamme de courbure

limite pour éviter l'aliasing spatial est (équ. (E2.7)):
$$q_{\text{range}} = \frac{1}{2f_{\text{max}}|x|_{\text{max}}\Delta x} = 125\text{ms/km}^2$$
. Or la

gamme effectivement employée est $q_{\text{max}} - q_{\text{min}} = 250 - (-100) = 350\text{ms/km}^2$. En principe on est en forte condition d'aliasing, mais deux effets se combinent ici pour réduire l'aliasing réel: d'une part en décomposition parabolique les pentes importantes -donc susceptibles d'être aliasées- ne se rencontrent que sur les grands offsets dans un domaine qui coïncide plus ou moins avec le maximum d'effet de stretch du NMO -qui réduit la fréquence de signal, donc l'aliasing-, et d'autre part c'est la courbure réelle d'un évènement par rapport aux bornes des courbures explorées qui détermine l'aliasing sur cet évènement (équ. (E2.12)). En fonction de cela les données sont en fait peu aliasées (fig. 7.4(a)).

La PRD en norme L2 est contaminée par un peu d'aliasing (fig. 7.4(b)), mais le résultat de l'interpolation est très bon (fig. 7.4(c)), ce que l'on vérifie en faisant la différence avec la solution exacte, seuls les grands offsets les plus susceptibles d'être aliasés montrant des erreurs un peu plus importantes (fig. 7.4(d)).

Si maintenant on part des mêmes données, mais de plus haute fréquence (ricker de fréquence centrale égale à 80Hz au lieu de 40Hz) (fig. 7.5(a)), la PRD en norme L2 est fortement contaminée par l'aliasing spatial (fig. 7.5(b)), avec comme conséquence une très mauvaise interpolation sur les nouvelles traces (fig. 7.5(c-d)). Une PRD parcimonieuse, ici en norme L1 sur le modèle, élimine pratiquement ce problème d'aliasing, toutes les traces étant correctement interpolées (fig. 7.5(e-f-g)).

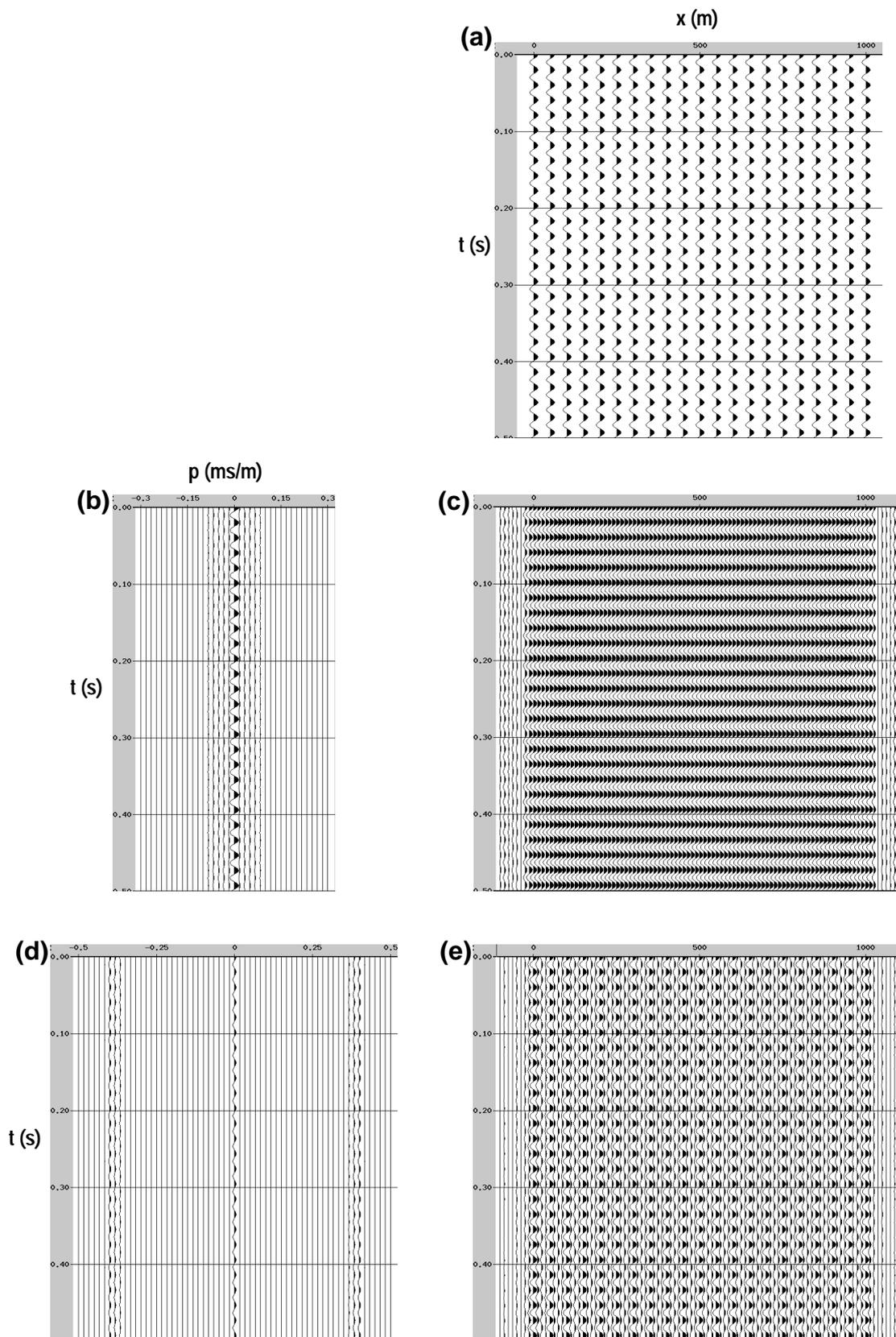


Figure 7.2 : (a) données monofréquentielles $f=50\text{Hz}$, $x_{min}=0$, $x_{max}=1000\text{m}$, $\Delta x=50\text{m}$. (b) (d) LRD en norme L2 avec $p_{max}=0.3\text{ms/m}$ et 0.5ms/m . (c) (e) reconstructions à partir de (b) et (d) avec $x_{min}=-100\text{m}$, $x_{max}=1100\text{m}$, $\Delta x=12.5\text{m}$.

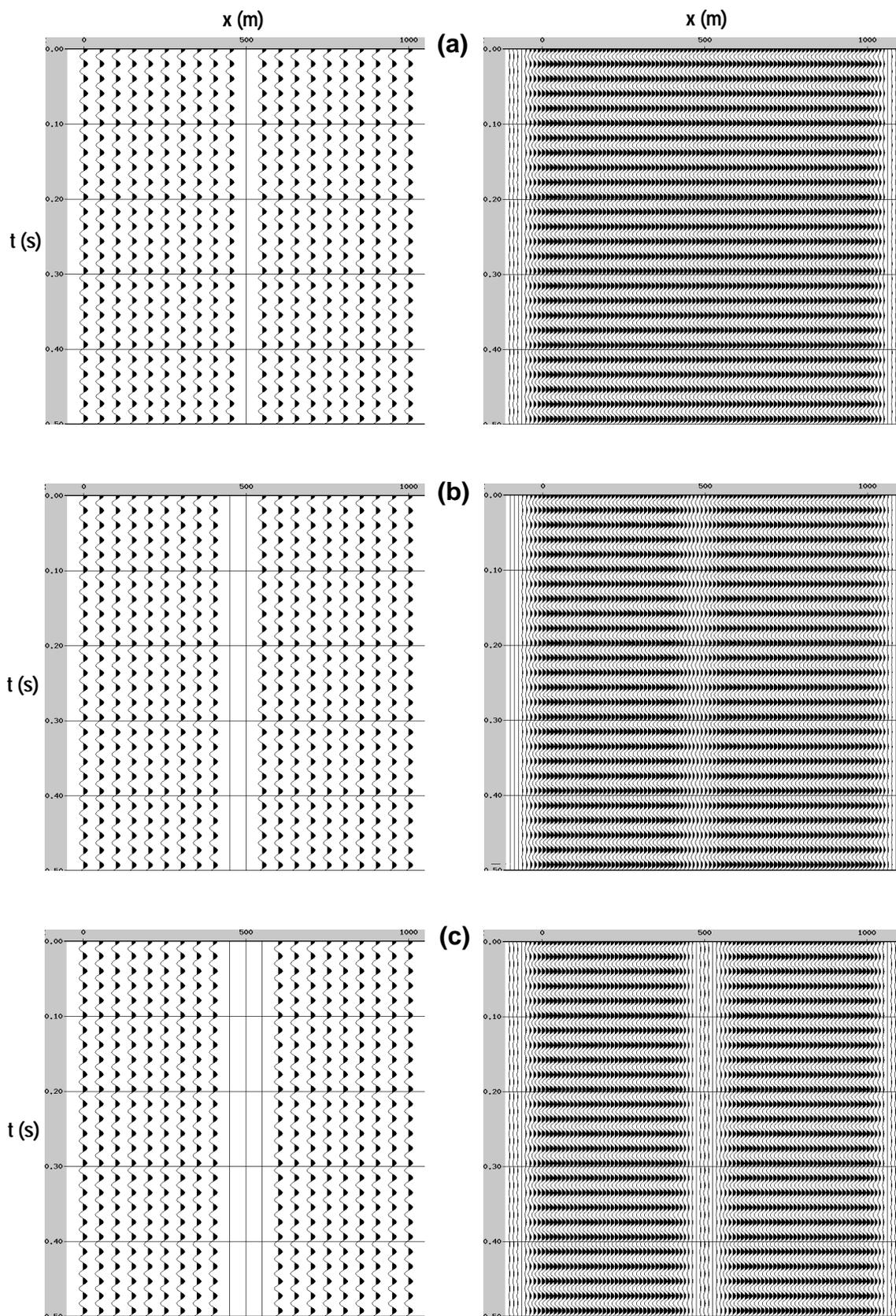
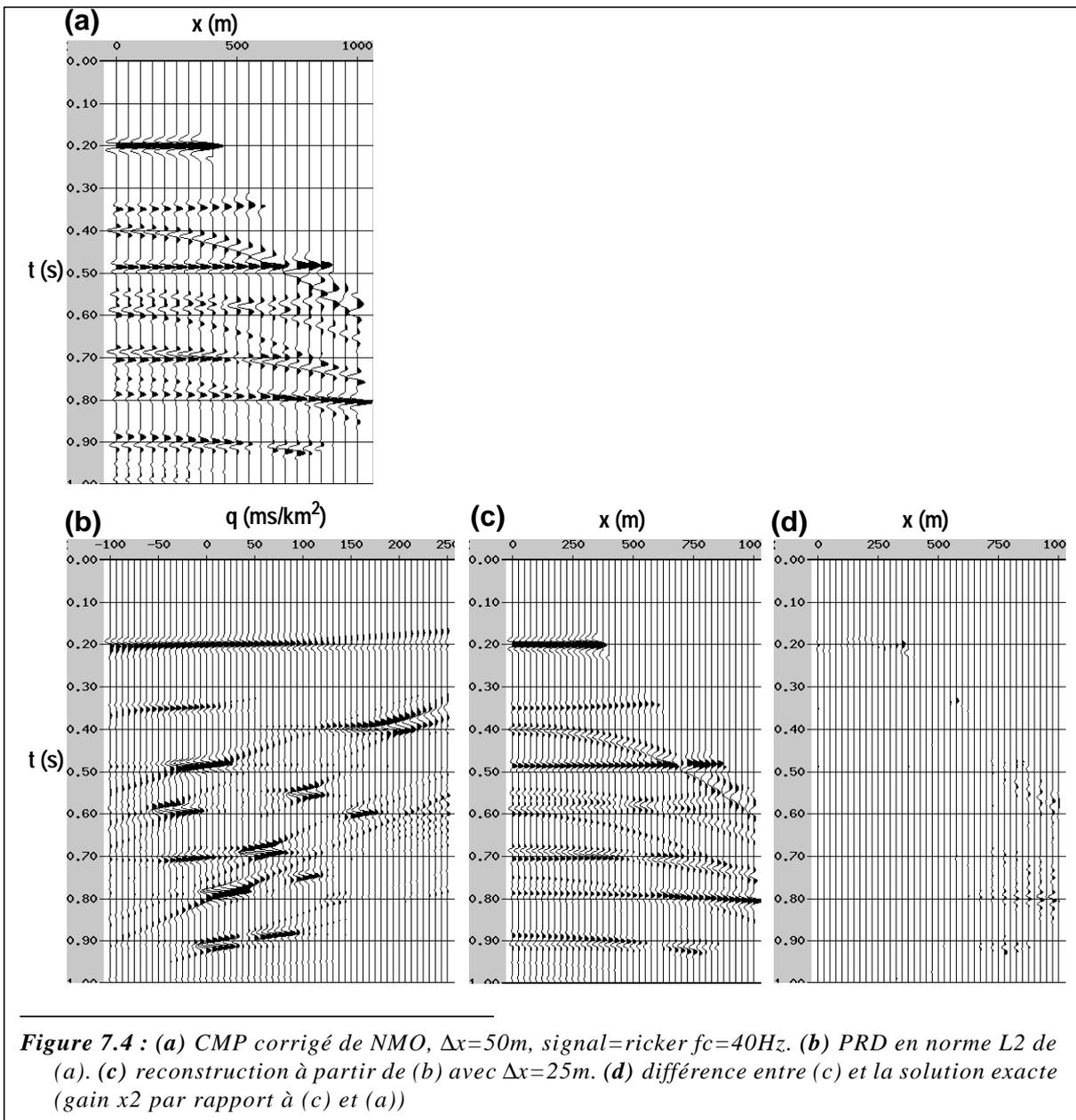


Figure 7.3 : (gauche) données fig. 7.2a, avec 1 (a), 2 (b), et 3 (c) traces manquantes, soit des gaps de 100, 150, et 200m. (droite) reconstructions après LRD norme L2 ($p_{max}=0.15ms/m$) avec $x_{min}=-100m$, $x_{max}=1100m$, $\Delta x=12.5m$.



Sur un deuxième test à partir de ces mêmes données (ricker 80Hz, intertrace $\Delta x=25m$), on retire des groupes de traces sur les offsets proches, lointains, et intermédiaires afin de tester l'extrapolation et l'interpolation à travers un gap (de tailles 50m, 100m, 150m, et 200m, fig. 7.6). Avec la PRD norme L1, la qualité de l'interpolation sur 200m est légèrement supérieure à celle obtenue avec la PRD norme L2 sur 100m. Ce type de test montre en général une interpolation possible sur des distances 3 fois plus grandes en utilisant la norme L1 par rapport à la norme L2. L'extrapolation est également meilleure avec la norme L1, mais reste très imparfaite. On note au passage que l'extrapolation des offsets proches est plus facile que celle des offsets lointains: en décomposition parabolique, toutes les paraboles s'aplatissent aux abords de l'offset nul, diminuant d'autant les problèmes d'aliasing (la relation (E2.6) montre d'ailleurs que le pas d'échantillonnage limite est d'autant plus grand que l'offset considéré est faible, signifiant que les offsets faibles sont en général sur-échantillonnés (le pas d'échantillonnage limite à $x=100m$ pour la fréquence centrale du ricker $f_c=80Hz$ vaut ici 179m).

Cette meilleure performance de la norme L1 à travers de larges gaps n'est rendue possible que parce que les événements sont proches des courbes utilisées pour la décomposition (des paraboles). Si on utilise des LRD au lieu des PRD sur le dernier cas (gap de 200m), la norme L1 n'améliore fortement le résultat que sur les événements les moins courbés (donc les plus proches de droites, courbes utilisées pour la décomposition) (fig. 7.7). Le problème est en effet de connecter 2 courbes qui ont des paramètres

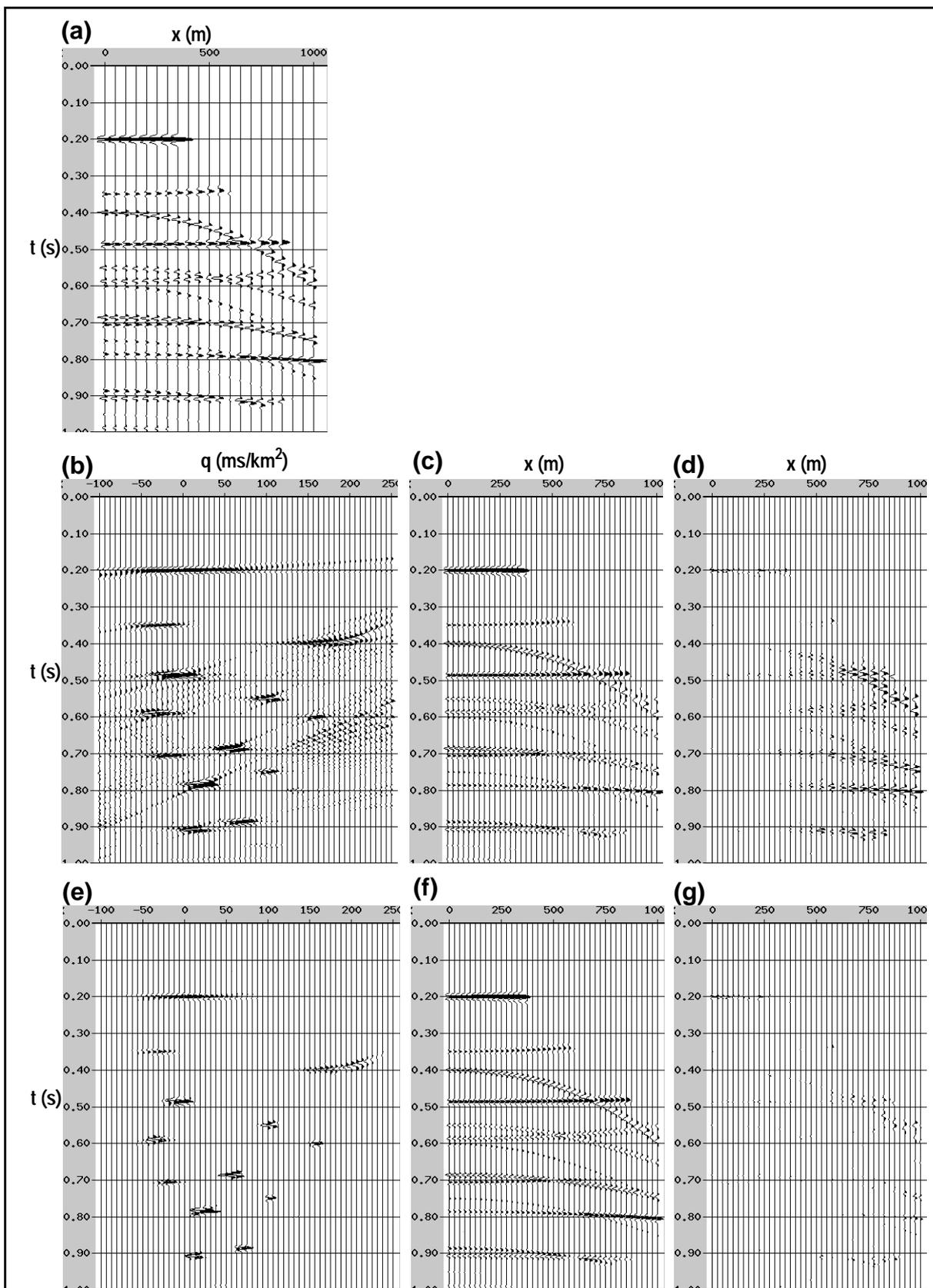


Figure 7.5 : (a) même CMP fig. 7.4(a) signal=ricker $f_c=80\text{Hz}$. (b) PRD en norme L2 de (a). (c) reconstruction à partir de (b) avec $\Delta x=25\text{m}$. (d) différence entre (c) et la solution exacte (gain $\times 2$ par rapport à (c) et (a)). (e) (f) (g) Idem (b) (c) (d) avec une PRD en norme L1 sur le modèle.

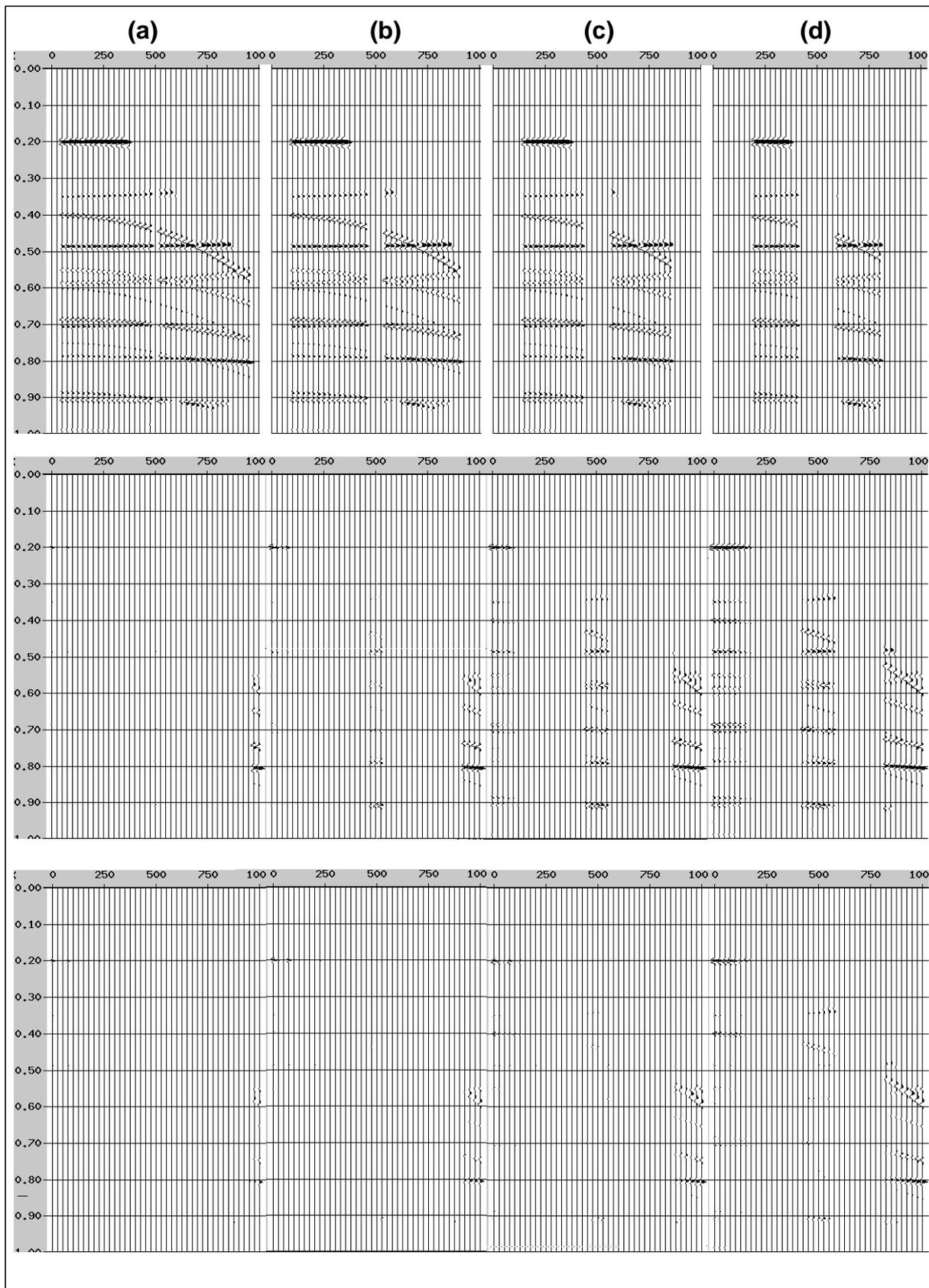


Figure 7.6 : (haut) données identiques à fig. 7.5a mais avec $\Delta x=25m$, et des traces raidies aux offsets proches, lointains et intermédiaires: (a) gap=50m (b) 100m (c) 150m (d) 200m. (milieu) erreur après interpolation et extrapolation des CMP en (haut) par PRD en norme L2. (bas) idem (milieu) par PRD en norme L1.

(pente dans le cas présent) différents de part et d'autre du gap (fig. 7.7). Les LRD sont effectués dans cet exemple avec une gamme de pente $-0.1 < p < 0.4 \text{ ms/m}$.

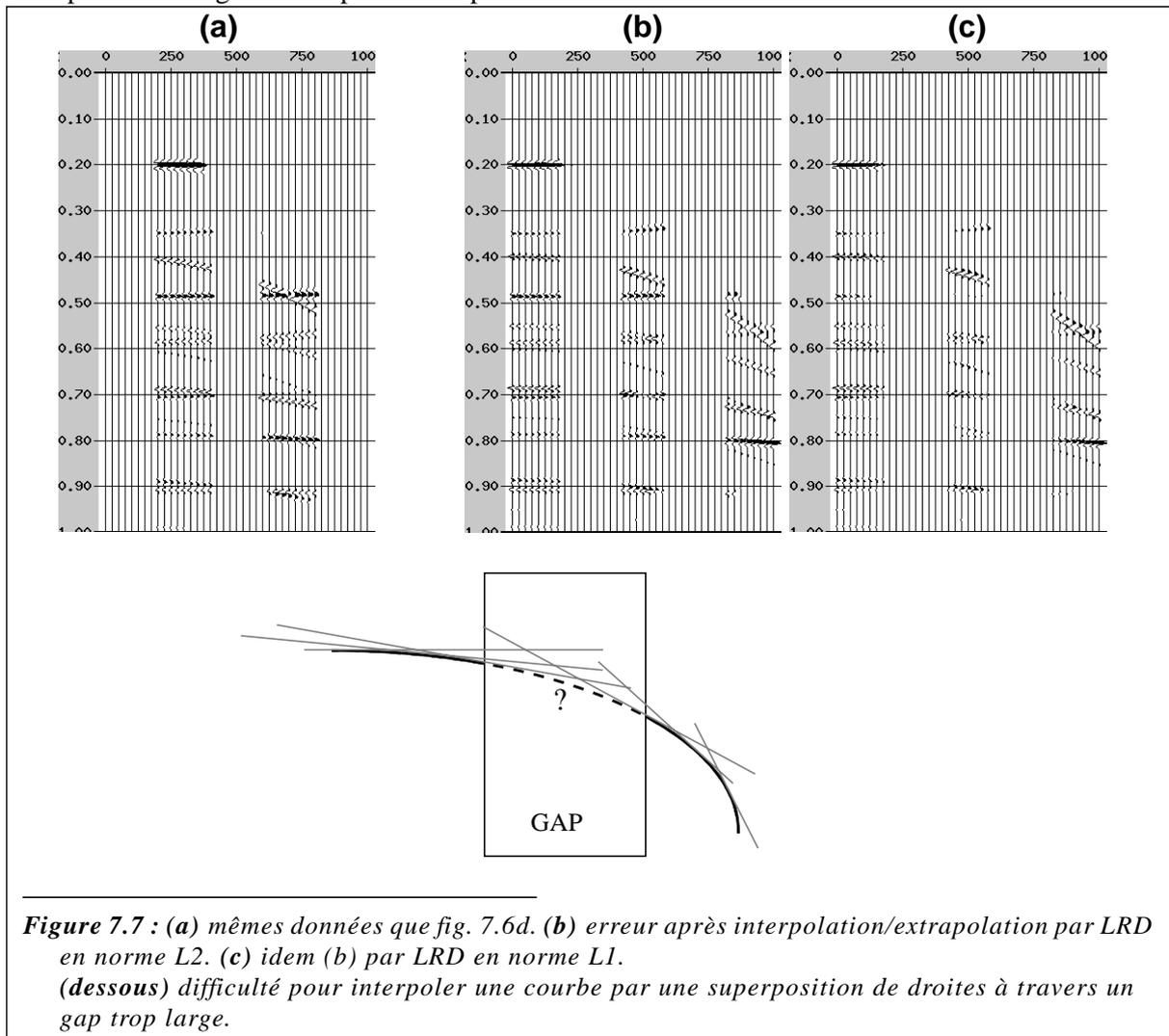


Figure 7.7 : (a) mêmes données que fig. 7.6d. (b) erreur après interpolation/extrapolation par LRD en norme L2. (c) idem (b) par LRD en norme L1. (dessous) difficulté pour interpoler une courbe par une superposition de droites à travers un gap trop large.

Les LRD peuvent virtuellement décomposer des données contenant tout type d'évènements (courbes, d'amplitudes variables, etc.), mais ce test souligne la nécessité, lorsque l'on utilise des décompositions parcimonieuses, de travailler sur des fenêtres spatiales dans lesquelles les évènements sont assimilables à des droites d'amplitude constante. En corollaire on ne peut évidemment pas interpoler au travers de gaps de tailles comparables aux fenêtres choisies.

7.4 - Interpolation 3D

7.4.1 - Avantages apportés par une direction spatiale supplémentaire

Lorsque les données sont 3D et disposées sur une grille irrégulière, la question ne se pose pas de savoir si il faut travailler en 3D, car il n'existe tout simplement pas de sous-domaines 2D dans lesquels on pourrait travailler, au contraire d'une grille régulière. Dans ce dernier cas, quel est l'avantage de travailler en 3D ? Nous avons vu (figure 6.3) que le passage de 3D à 2D peut introduire artificiellement de l'aliasing, dès lors que le support des spectres spatiaux (ou des pentes) n'est pas rectangulaire. De manière plus générale on peut raisonnablement supposer que toute dimension supplémentaire est susceptible de réduire l'aliasing spatial. Ainsi sur une grille régulière une trace manquante à reconstruire est encadrée par deux traces seulement en 2D, alors qu'elle est entourée de 8 voisins en 3D (ou 6 si la grille est en quinconce), ce qui intuitivement ne peut que faciliter l'interpolation.

7.4.2 - Méthodologie

Les données consistent en un ensemble de traces distribuées de manière quelconque dans un espace (x,y) , où x et y sont deux variables spatiales.

Il n'est bien sûr pas question d'effectuer une décomposition de Radon globale sur l'ensemble des données! D'une part pour une raison de coût CPU et d'autre part parce que les événements réels pris sur un trop grand domaine n'ont rien à voir avec des plans d'amplitudes constantes, hypothèse de base si on veut utiliser les décompositions parcimonieuses. On découpe donc le domaine en vignettes de taille à définir, et chaque vignette est traitée indépendamment des autres: LRD 3D locale et reconstruction sur la grille désirée à l'intérieur de la vignette. On introduit des recouvrements entre les vignettes, et donc des rampes (linéaires par exemple) sur les traces reconstruites (voir figure 7.8).

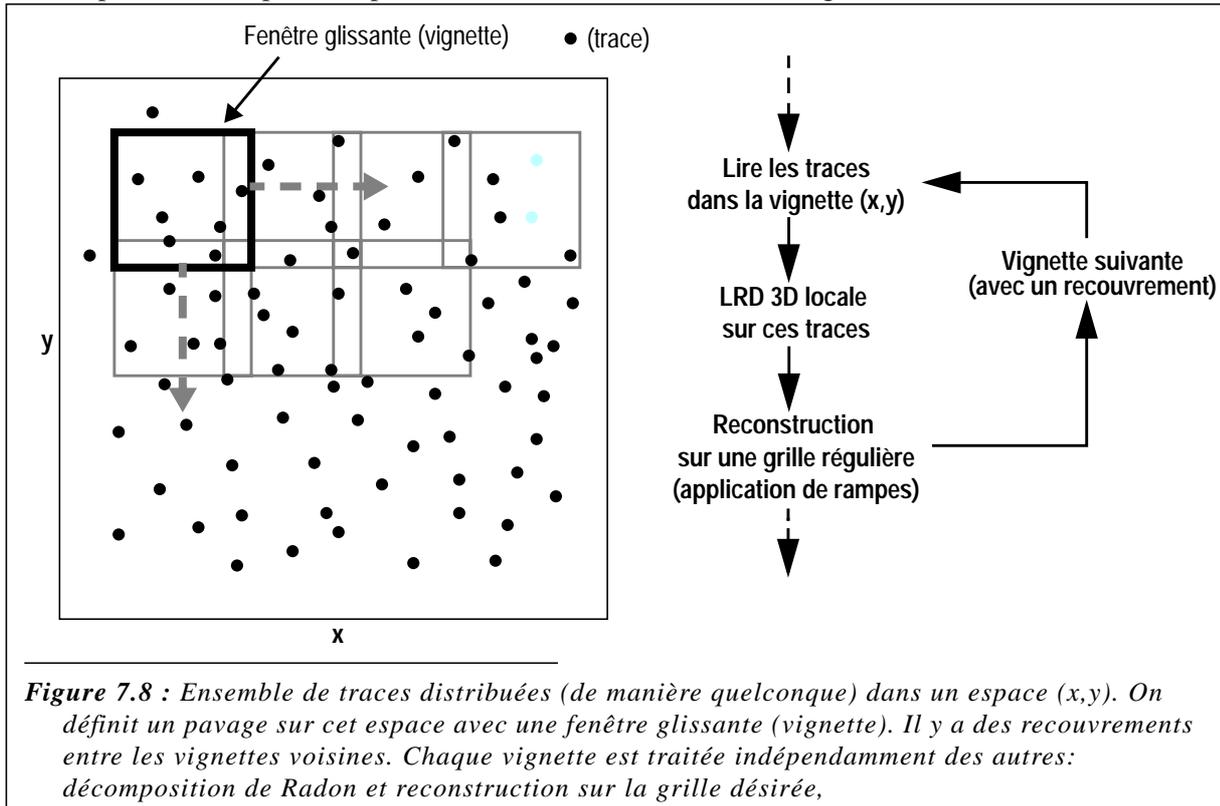


Figure 7.8 : Ensemble de traces distribuées (de manière quelconque) dans un espace (x,y) . On définit un pavage sur cet espace avec une fenêtre glissante (vignette). Il y a des recouvrements entre les vignettes voisines. Chaque vignette est traitée indépendamment des autres: décomposition de Radon et reconstruction sur la grille désirée,

Concernant le temps de calcul, supposons que les données contiennent $N=N_x.N_y$ traces régulièrement réparties, et que l'on divise le domaine en vignettes de $n=n_x.n_y$ traces (le nombre de vignettes est donc $(N_x.N_y)/(n_x.n_y)=N/n$). Une fois fixées les gammes de pentes explorées, et pour une géométrie donnée, le nombre de traces du modèle est directement proportionnel à la taille de la vignette, donc au nombre de traces dans la vignette: $N_p=\alpha.n$. Le nombre d'opérations pour la décomposition et la reconstruction pour une vignette est de l'ordre de $N_p.n=\alpha.n^2$, ce qui donne pour tout le domaine (si il n'y a pas de recouvrement) $(\alpha.n^2).(N/n)=\alpha.n.N$. Le temps de calcul total est donc proportionnel à la taille choisie pour la vignette.

La taille de la vignette est donc un compromis entre trois effets:

- Pour le temps de calcul, on a intérêt à choisir la vignette la plus petite possible
- Pour approximer les événements par des plans d'amplitudes constantes (pour les décompositions parcimonieuses), on a intérêt à choisir la vignette la plus petite possible
- Une vignette trop petite ne contient plus assez de traces, avec deux conséquences: les effets de bords de la décomposition de Radon deviennent prépondérants, et la résolution en pentes est très mauvaise.

Les recouvrements sont en fait introduits pour tenir compte des effets de bord. Si la taille du recouvrement est de $m=m_x.m_y$ traces, le nombre de vignettes nécessaire est égal à $N/(n-m)$, et le temps

total de calcul sur tout le domaine de l'ordre de $\alpha.n^2.N/(n-m)$ (étant entendu que $n>m$). La taille optimum de la vignette pour un recouvrement m fixé est alors $n=2.m$, pour un temps de l'ordre de $\alpha.m.N$. La taille optimum (du point de vue temps de calcul) de la vignette est donc le double de la taille du recouvrement.

7.4.3 - Description des données synthétiques de test

Les données utilisées pour les tests consistent en une ligne 2D. Les deux variables spatiales sont les abscisses sources et récepteurs.

On utilise un programme de modélisation acoustique par différences finies 2D. Il n'est possible que d'effectuer des campagnes de tir à géométrie régulière. On effectue donc une acquisition très serrée, avec une distance entre tirs et récepteurs de 5m, puis on la décime en enlevant aléatoirement 93.75% des traces dans un premier cas et 99% dans un second cas, afin de simuler deux acquisitions irrégulières, l'une avec une distance moyenne entre tirs et entre récepteurs de 20m (SI20), l'autre avec une distance moyenne de 50m (SI50). Les abscisses source et récepteur sont ainsi pseudo-aléatoires pour chaque trace (figure 7.9).

Les positions des sources vont de $x_s=2000m$ à $x_s=4000m$, les positions récepteurs allant de $x_r=x_s-2000m$ à $x_r=x_s-50m$. Sur les deux acquisitions SI20 et SI50, on enlève les traces dans un carré de 250m de côté ($1375 < x_r < 1625$, $2375 < x_s < 2625$)

La fréquence maximum de la source utilisée pour la modélisation est environ $f_{max}=50Hz$ (fig. 7.10).

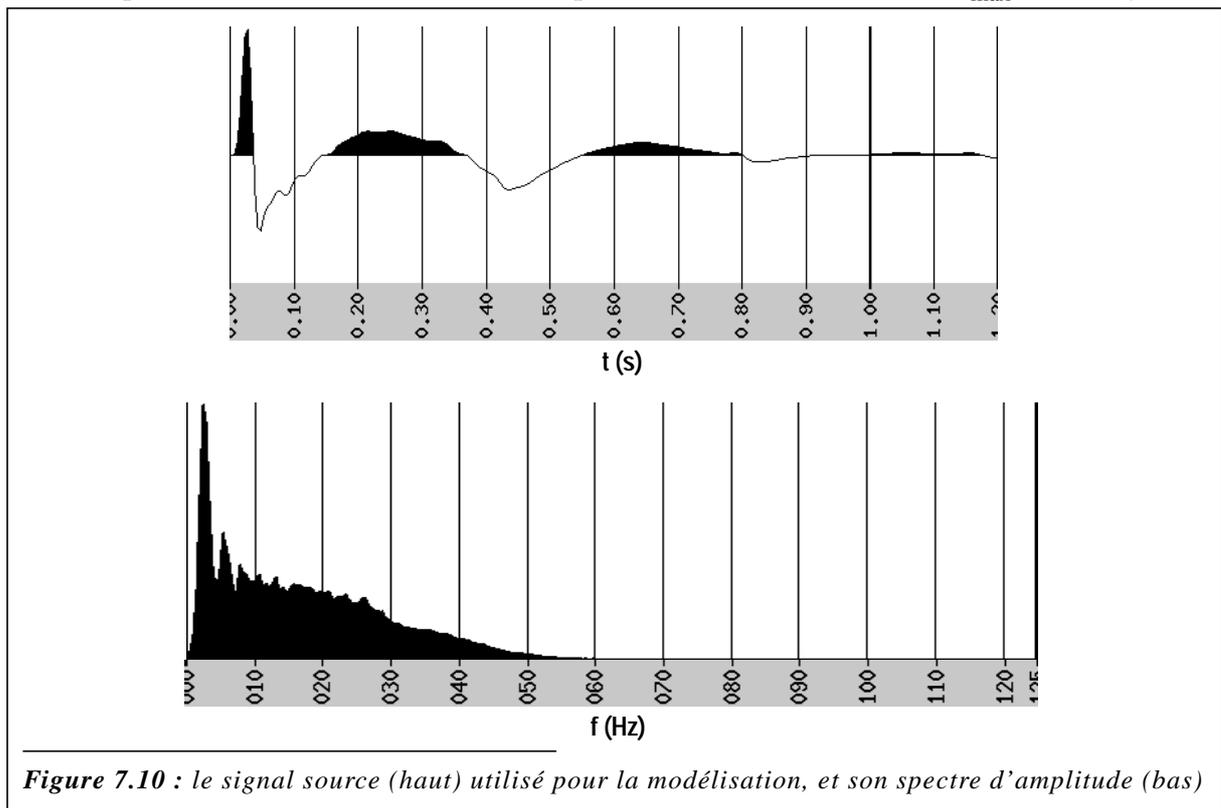


Figure 7.10 : le signal source (haut) utilisé pour la modélisation, et son spectre d'amplitude (bas)

On s'intéresse aux résultats de la régularisation de l'acquisition sur deux points de tirs (CSP1 et CSP2, voir fig. 7.9). La solution exacte constituée par les points de tir sur une grille régulière est tracée sur la figure 7.11.

Les gammes de pentes utilisées pour les décompositions de Radon sont:

$$\begin{aligned} -0.7 \leq p_{xr} \leq 0.3 \text{ms/m} \\ -0.3 \leq p_{xs} \leq 0.7 \text{ms/m} \end{aligned} \tag{E7.2}$$

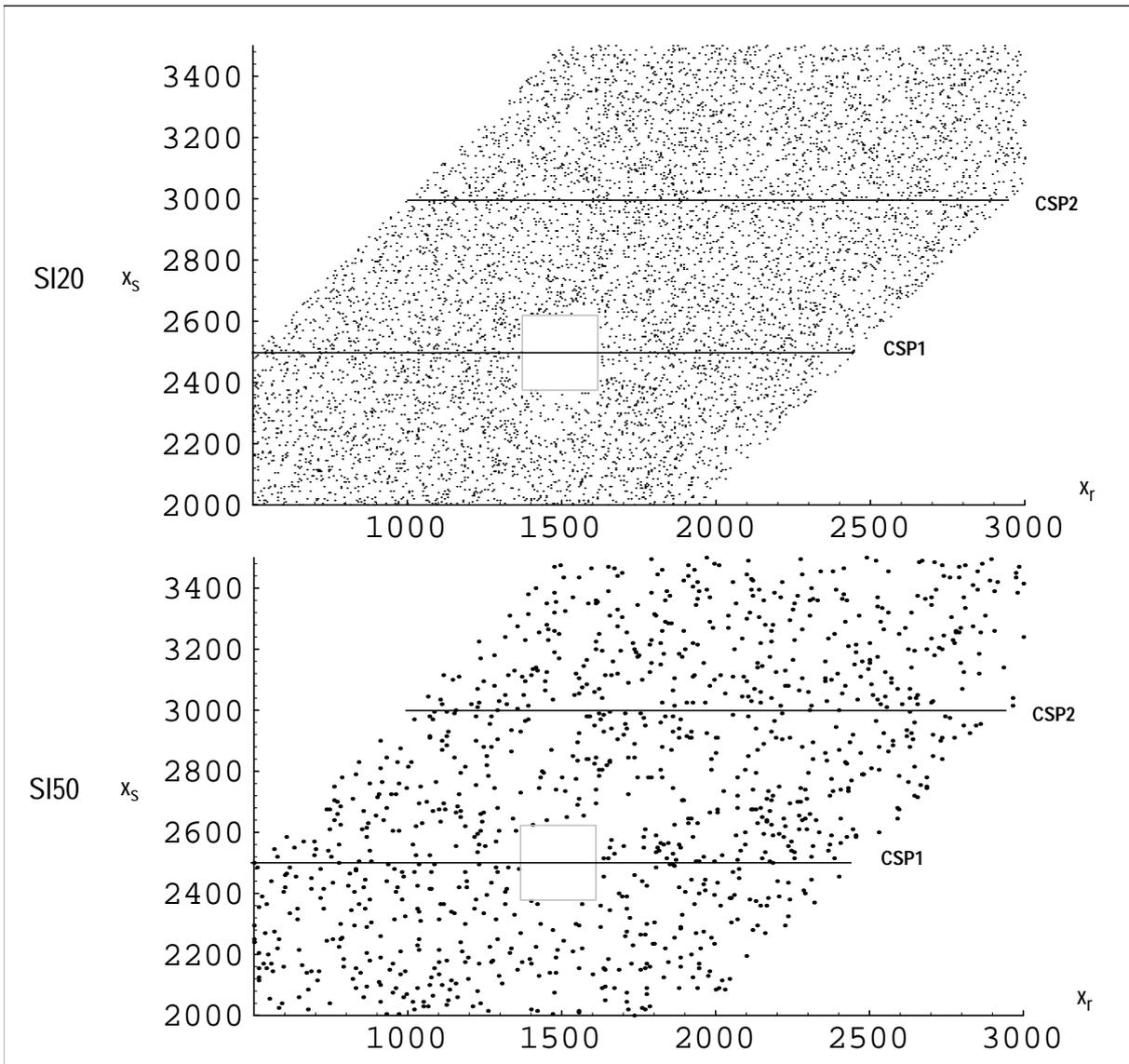


Figure 7.9 : Deux acquisitions irrégulières, où les abscisses sources récepteurs sont pseudo-aléatoires pour chaque trace. Distances moyennes entre sources et entre récepteurs: 20m (SI20) et 50m (SI50). Les traces ont été enlevées dans un carré de 250x250m² pour simuler un gap. CSP1 et CSP2 sont deux points de tirs sur lesquels on teste la régularisation de l'acquisition.

Les pas d'échantillonnage spatiaux limites à respecter pour éviter l'aliasing spatial sont donc (équ. (E2.5)):

$$\Delta x_s \leq (\Delta x_s)_{th} = \frac{1}{f_{\max}((p_{xs})_{\max} - (p_{xs})_{\min})} = 20m$$

$$\Delta x_r \leq (\Delta x_r)_{th} = \frac{1}{f_{\max}((p_{xr})_{\max} - (p_{xr})_{\min})} = 20m$$
(E7.3)

Ainsi, l'acquisition SI20 n'est pas aliasée *en moyenne*, alors que SI50 l'est fortement.

7.4.4 - Rééchantillonnage de données non-aliasées

On veut ici régulariser l'acquisition SI20. Le première tentative consiste en un *flex-binning* (on affecte au centre du bin la trace la plus proche; si aucune trace n'est présente dans le bin on duplique une trace

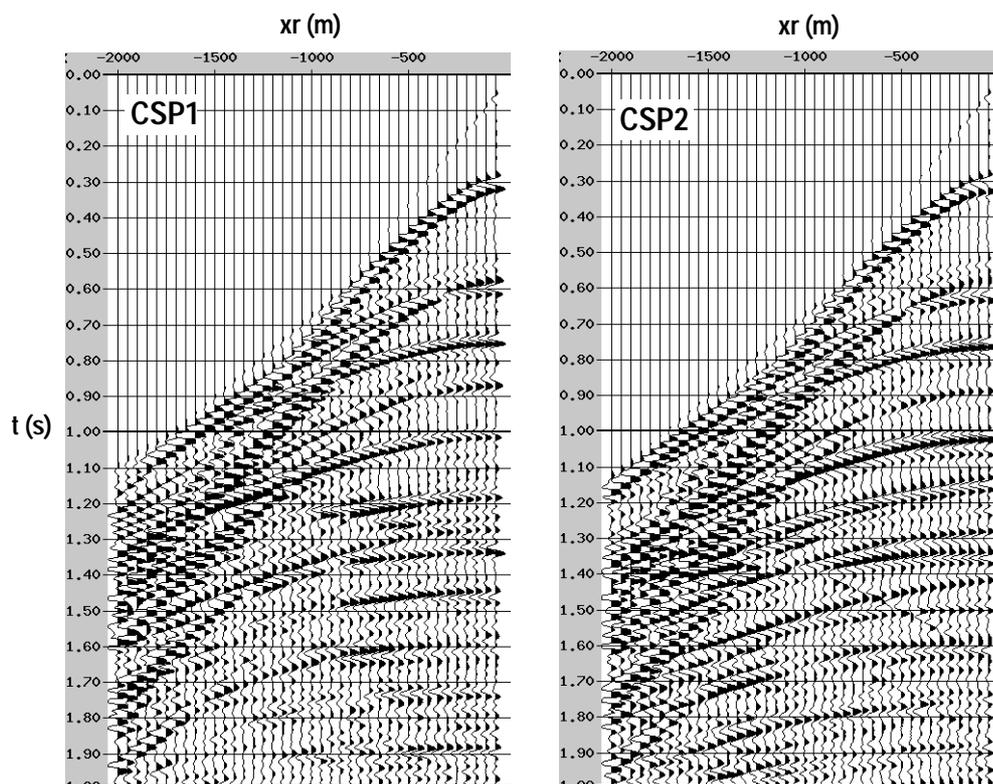


Figure 7.11 : Points de tir CSP1 et CSP2 échantillonnés sur une grille régulière.

depuis un bin voisin) (fig. 7.12, seule l'erreur est tracée, différence entre la solution exacte fig. 7.11 et le résultat). Le binning n'est évidemment pas satisfaisant dès lors que les évènements sont pentés.

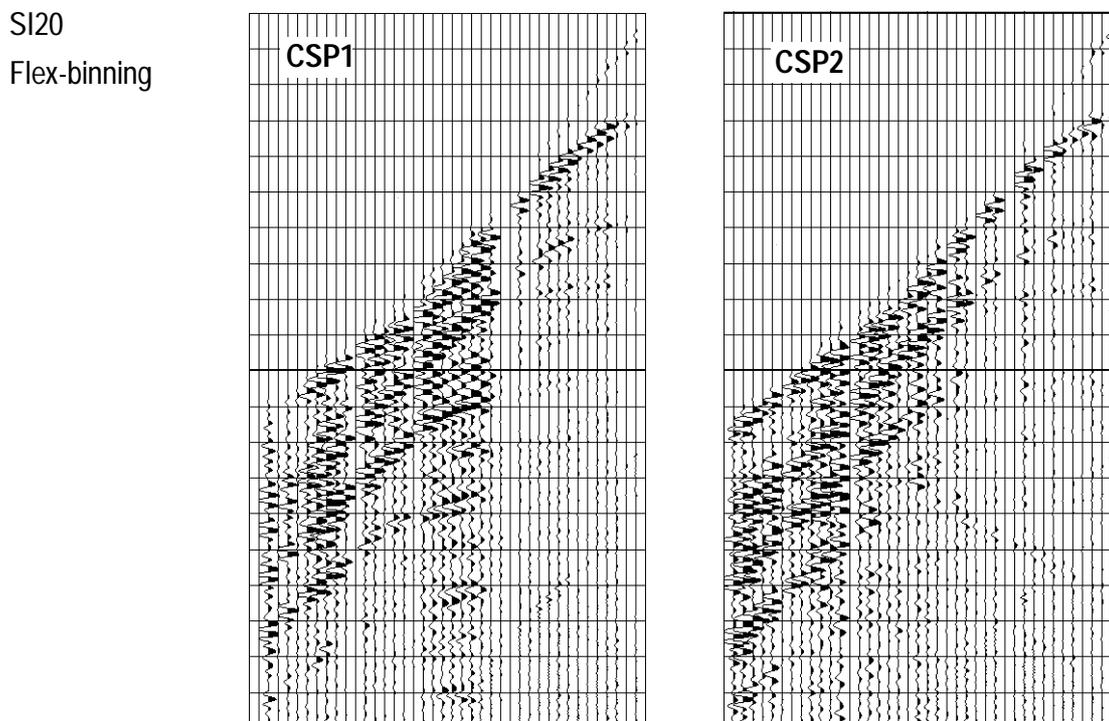


Figure 7.12 : Erreur sur régularisation de SI20 par flex-binning.

La régularisation par LRD 3D en norme L2 est par contre de très bonne qualité (fig. 7.13). Tous les évènements sont bien interpolés, y compris dans les zones où il y a des pentes conflictuelles. Deux tailles de vignette/recouvrement ont été utilisées: 250/100m et 500/250m. Les résultats sont très comparables,

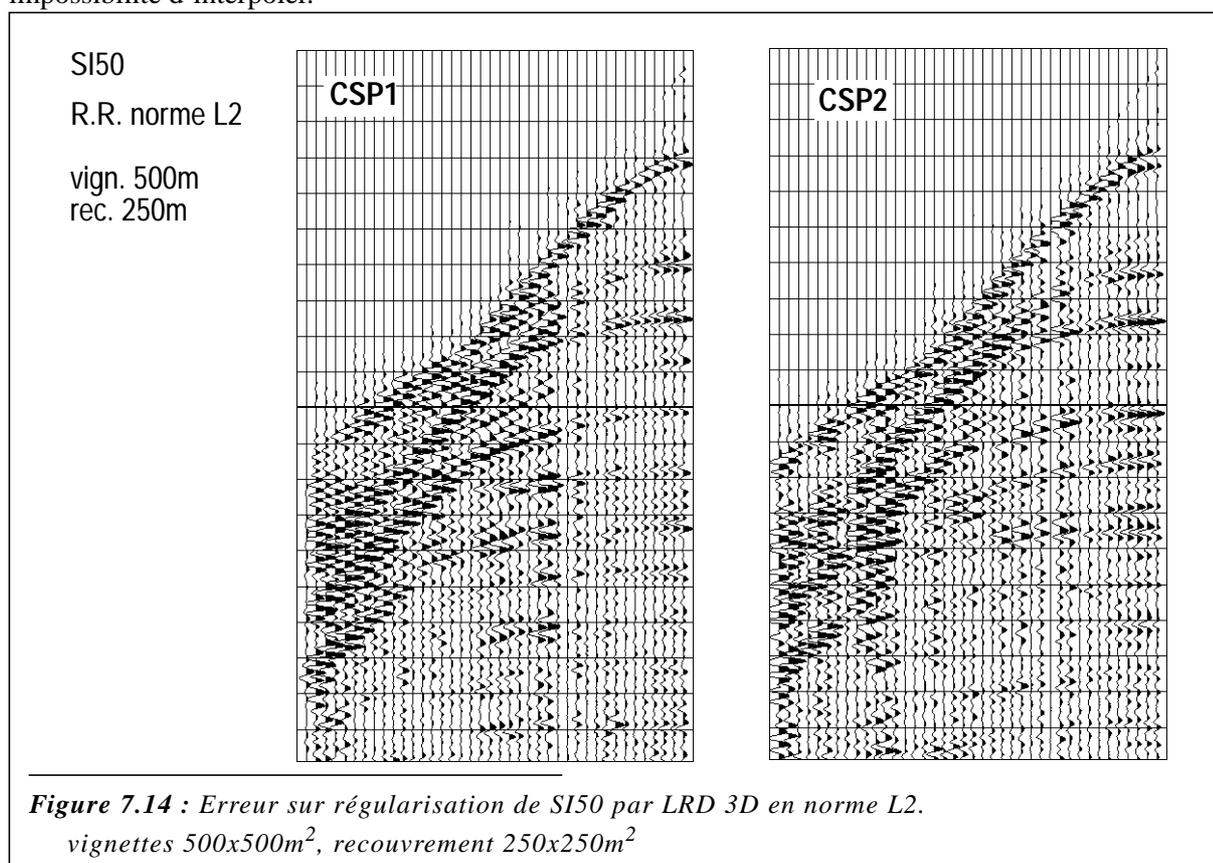
avec peut-être une qualité légèrement meilleure avec la grande vignette (mais avec un coût 4 fois plus élevé). Les seules traces pour lesquelles l'erreur est importante sont situées dans le gap ($250 \times 250 \text{m}^2$) de l'acquisition: rappelons qu'avec la norme L2, la distance d'interpolation possible est ici de l'ordre de 20m.

7.4.5 - Rééchantillonnage de données aliasées

On veut ici régulariser l'acquisition SI50, fortement aliasée spatialement.

7.4.5.1 - ...décompositions en norme L2...

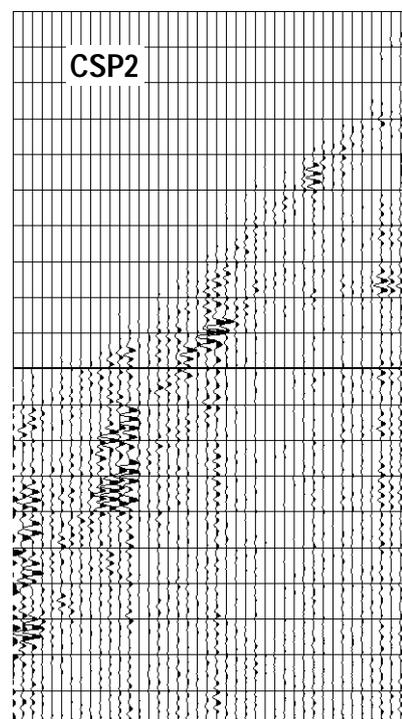
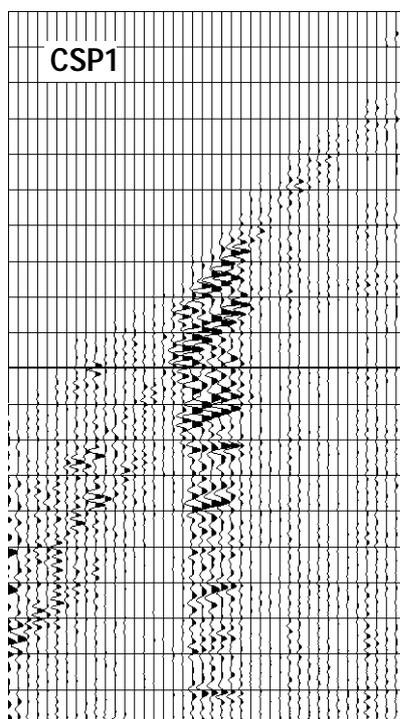
La régularisation par LRD 3D en norme L2 échoue complètement (fig. 7.13). L'amplitude de l'erreur est du même ordre que l'amplitude des données d'origine (fig. 7.11). L'aliasing est responsable de cette impossibilité d'interpoler.



7.4.5.2 - ...décompositions parcimonieuses...

L'utilisation des LRD parcimonieuses améliore nettement le résultat (fig. 7.13). Nous avons utilisé les LRD obtenues par norme L1 sur le modèle d'une part (IRLS type 1, 12 itérations dans chaque optimisation à poids constant, voir paragr. 4.2 et 3.5.3) et par estimation des rapports S/B d'autre part (3 séries de 5 itérations à poids constant, voir paragr. 4.4.2.1). Les deux méthodes donnent des résultats similaires (légèrement meilleurs avec l'estimation des rapports S/B), et de qualité à peine inférieure qu'avec les LRD 3D en norme L2 sur les données non aliasées (fig. 7.13). Il reste néanmoins certaines traces mal interpolées. Ces dernières sont situées dans des zones où la densité de traces dans les données est très faible, voire nulle.

SI20
R.R. norme L2
vign. 250m
rec. 100m



SI20
R.R. norme L2
vign. 500m
rec. 250m

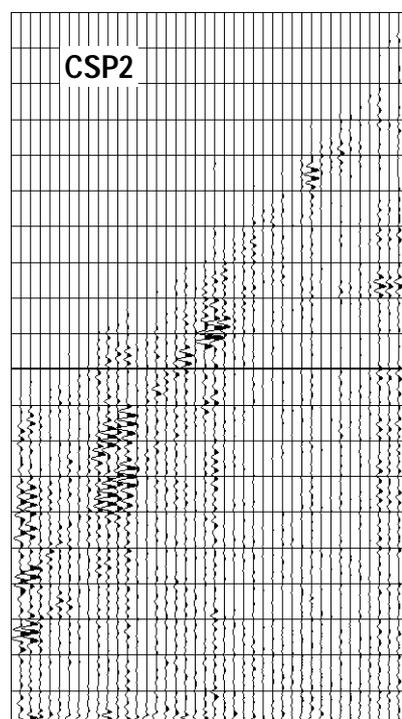
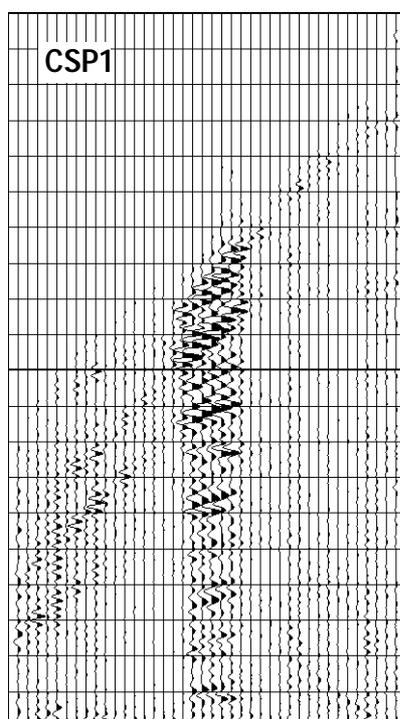
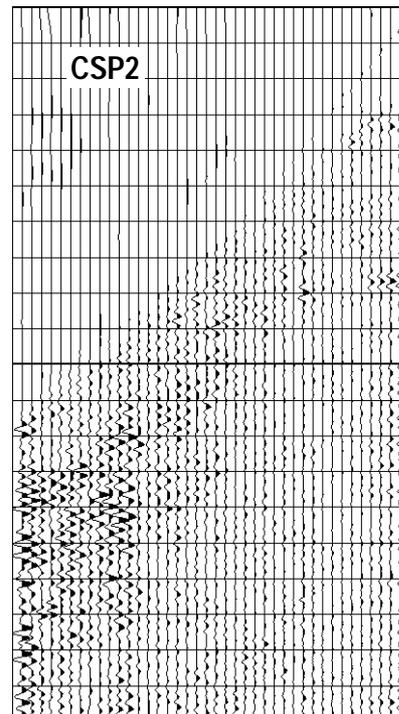
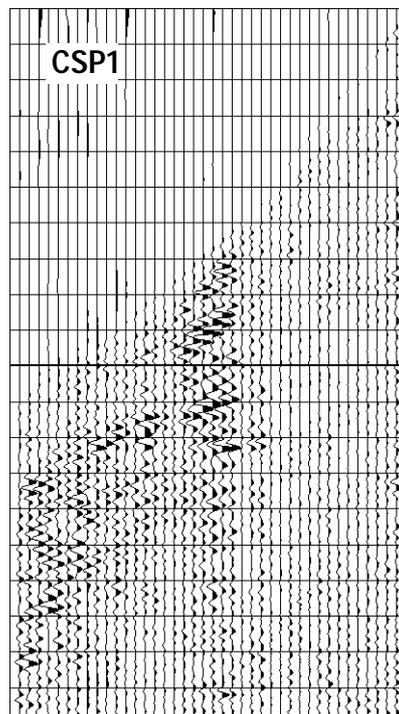


Figure 7.13 : Erreur sur régularisation de SI20 par LRD 3D en norme L2.

(haut) vignettes $250 \times 250 \text{m}^2$, recouvrement $100 \times 100 \text{m}^2$

(bas) vignettes $500 \times 500 \text{m}^2$, recouvrement $250 \times 250 \text{m}^2$

SI50
R.R. norme L1
vign. 500m
rec. 250m



SI50
R.R. S/B
vign. 500m
rec. 250m

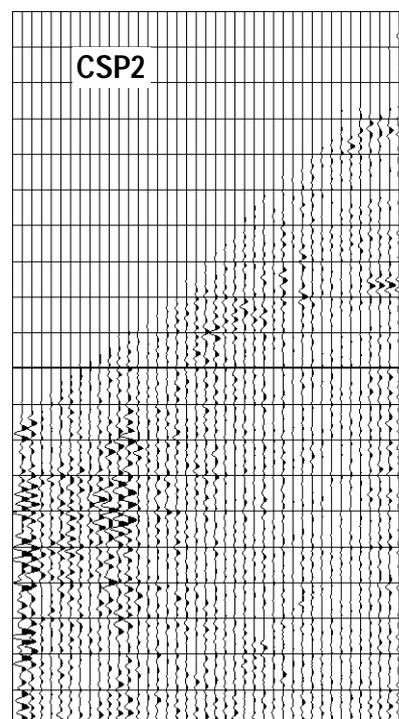
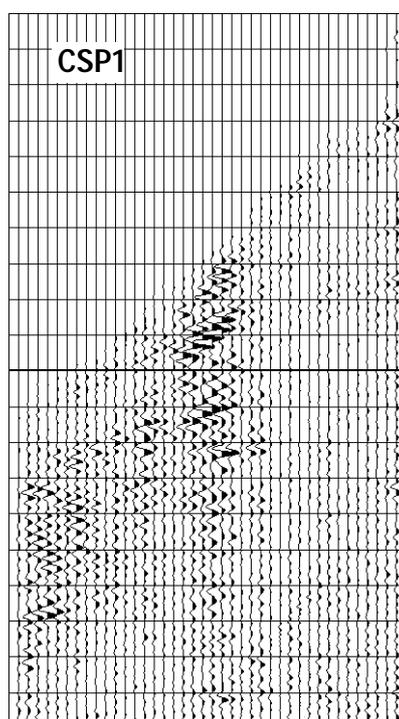
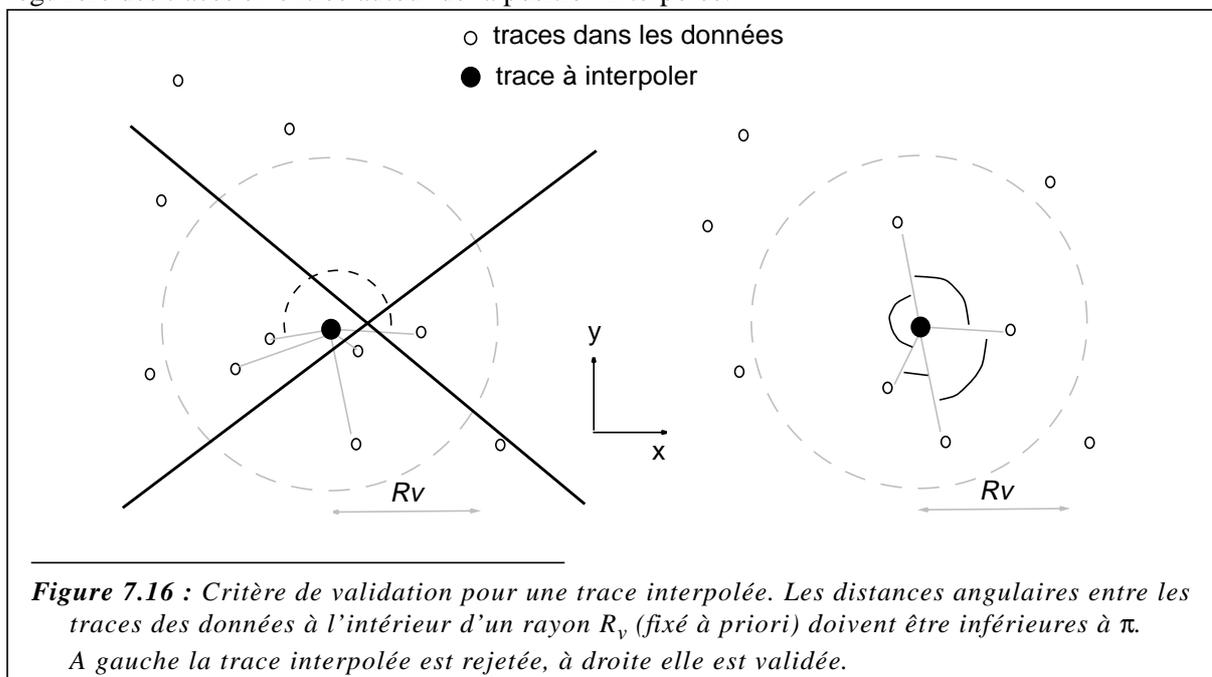


Figure 7.15 : Erreur sur régularisation de SI50 par LRD 3D en norme L1 (**haut**) et estimation des rapports S/B (**bas**).
vignettes 500x500m², recouvrement 250x250m²

7.4.5.3 - ...validation des traces interpolées...

On a donc besoin d'un critère pour valider ou rejeter une trace interpolée. On se base pour cela sur la répartition spatiale des traces autour de la position sur laquelle on veut interpoler. On fixe un rayon R_v autour de la trace à interpoler définissant un disque, et on considère toutes les traces des données à l'intérieur du disque. La distance angulaire entre deux traces consécutives doit être inférieure à π pour que l'interpolation soit validée (fig. 7.16). Ce critère géométrique assure une distribution spatiale régulière des traces en entrée autour de la position interpolée.



Toute la qualité du critère réside bien sûr dans le choix du rayon. Si la valeur choisie est trop grande on risque de conserver des traces mal interpolées, et si elle est trop faible on risque d'éliminer trop de traces. On teste ce critère sur la régularisation par LRD 3D en norme L1 de la figure 7.13, avec plusieurs valeurs de R_v : 80m, 60m, et 40m (fig. 7.17). A 80m on rejette la plupart des mauvaises traces interpolées, tandis qu'à 40m on rejette visiblement des traces qui semblaient bien interpolées. 60m semble être un compromis correct.

De manière générale, les quelques tests effectués montrent bien sûr que le rayon optimum dépend du pas d'échantillonnage théorique à respecter (équ. (E7.3)). Avec les LRD en norme L2, nous avons en fait déjà vu que le rayon devait être de l'ordre de ce pas d'échantillonnage. Avec les LRD parcimonieuses utilisées ici, on peut monter jusqu'à environ 3 fois ce pas.

LRD norme L2	$R_v \approx (\Delta x)_{th}$	(E7.4)
LRD parcimonieuses	$R_v \approx 3(\Delta x)_{th}$	

Ces relations sont empiriques. Notons bien qu'il s'agit d'un rayon, ce qui signifie que l'on peut au maximum interpoler au travers de gaps de largeur ($2R_v$), si il y a suffisamment de traces sur les bords.

Une fois fixé le rayon R_v , il est recommandé de choisir un recouvrement au moins égal à $2R_v$, afin que toute trace interpolée soit à l'intérieur d'au moins une vignette qui contienne tout le disque.

7.4.5.4 - ...influence de la taille des vignettes...

Nous avons vu que la taille des vignettes influe peu sur les résultats obtenus par LRD en moindres carrés. Il en va autrement avec les décompositions parcimonieuses: une petite vignette permet en

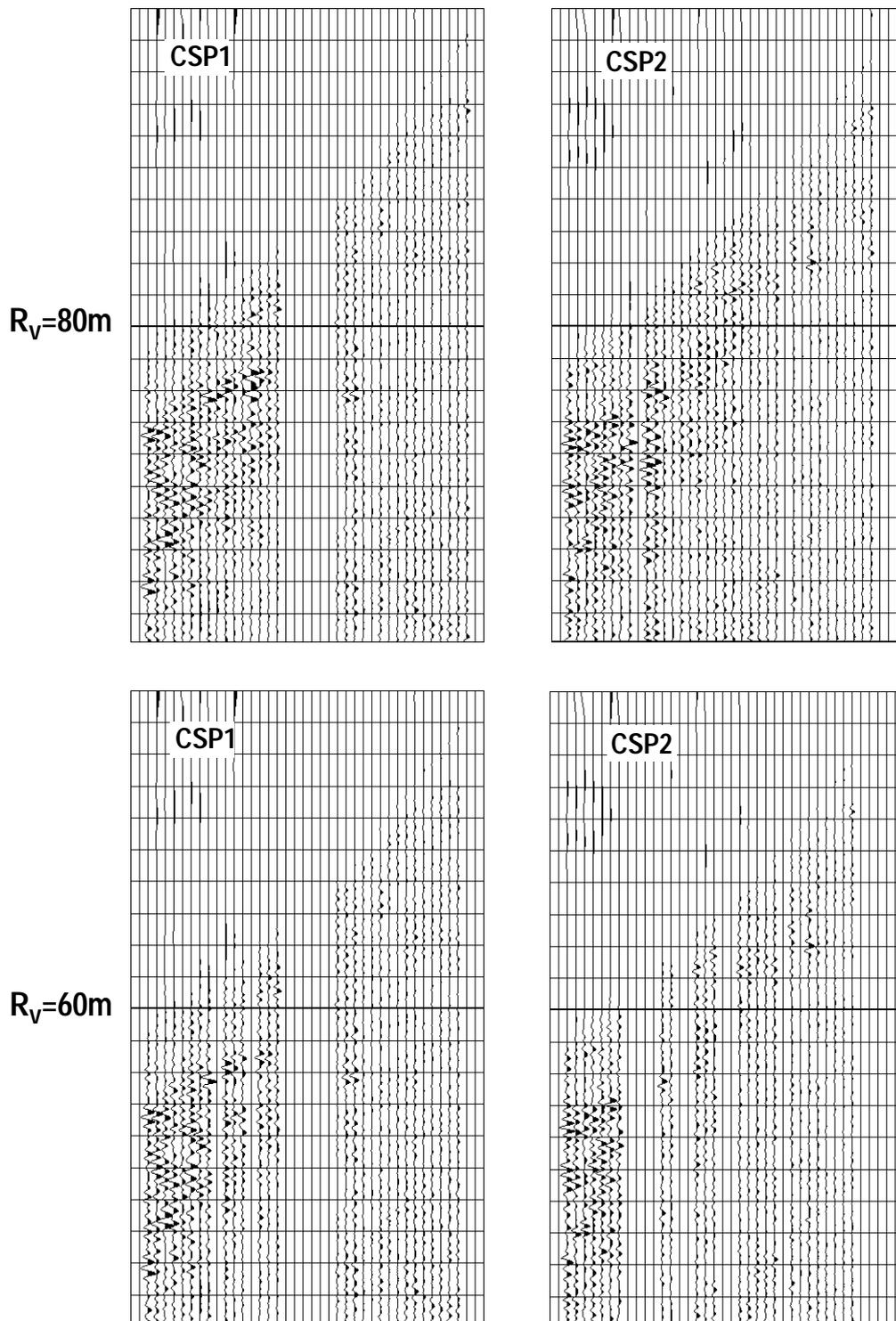
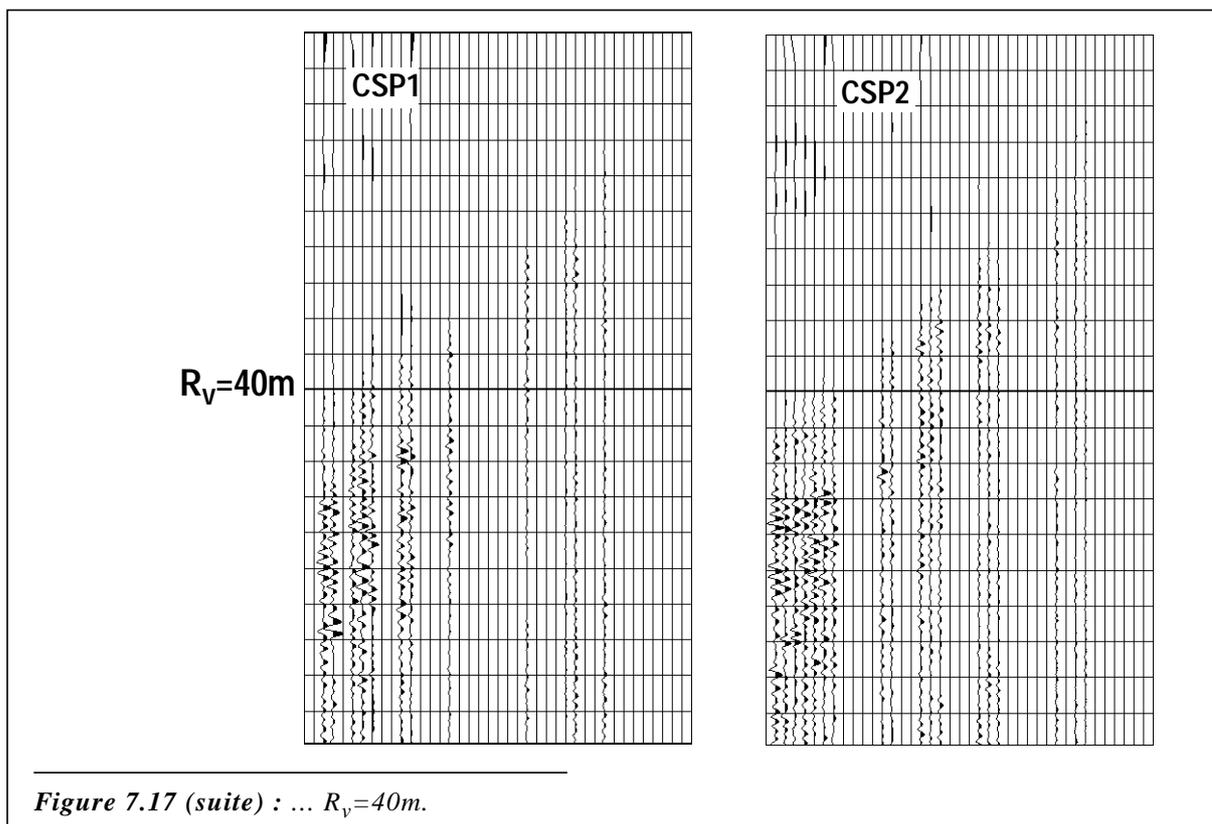


Figure 7.17 : Application du critère de validation (fig. 7.16) sur le résultat par LRD norme L1 (fig. 7.13 haut). Les traces rejetées sont raidies. (**haut**) $R_v=80m$. (**bas**) $R_v=60m$.



principe de mieux approximer les événements par des plans d'amplitudes constantes, mais réduit de fait la taille des gaps au travers desquels on peut interpoler.

Dans les deux tests avec décompositions parcimonieuses (fig. 7.13), la taille des vignettes est $500 \times 500 \text{m}^2$, avec un recouvrement de $250 \times 250 \text{m}^2$.

On effectue les mêmes tests avec des vignettes de $250 \times 250 \text{m}^2$, recouvrement $100 \times 100 \text{m}^2$ (fig. 7.13), et de $1000 \times 1000 \text{m}^2$, recouvrement $500 \times 500 \text{m}^2$ (fig. 7.13). Le résultat obtenu avec les vignettes de 250m est un peu moins bon que celui obtenu avec celles de 500m . Celui obtenu avec les vignettes de 1000m est meilleur sur certains événements (les événements presque linéaires aux grands offsets), et moins bon sur d'autres (les événements courbés aux faibles offsets). La vignette de 500m choisie au départ semble être un bon compromis.

On retrouve ces constatations sur les tableaux ci-dessous résumant pour les deux points de tir l'énergie des erreurs (en pourcentage par rapport à l'énergie de la solution exacte).

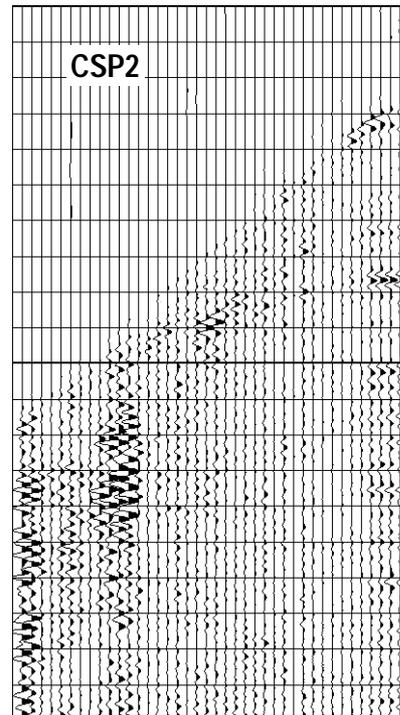
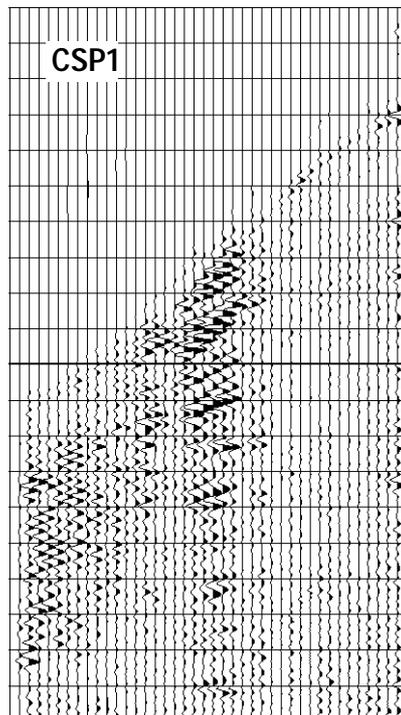
Tableau 4 : pourcentage d'erreur sur régularisation de CSP1

LRD\vignette	$250 \times 250 \text{m}^2$	$500 \times 500 \text{m}^2$	$1000 \times 1000 \text{m}^2$
norme L2	53%	49%	
norme L1	13.0%	6.9%	6.6%
S/B	13.6%	6.3%	5.7%

Tableau 5 : pourcentage d'erreur sur régularisation de CSP2

LRD\vignette	$250 \times 250 \text{m}^2$	$500 \times 500 \text{m}^2$	$1000 \times 1000 \text{m}^2$
norme L2	51%	50%	
norme L1	7.2%	4.5%	4,9%

SI50
R.R. norme L1
vign. 250m
rec. 100m



SI50
R.R. S/B
vign. 250m
rec. 100m

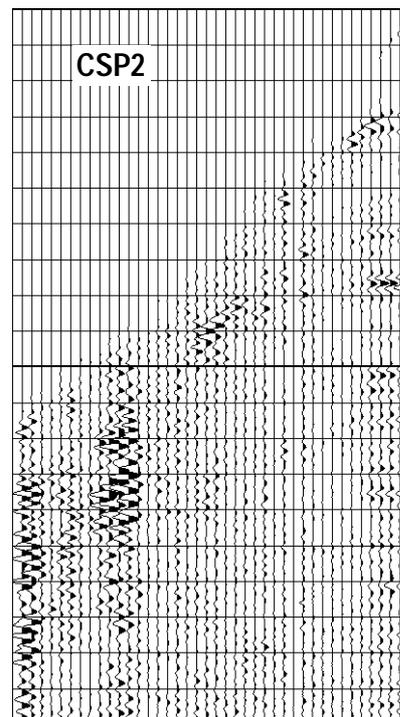
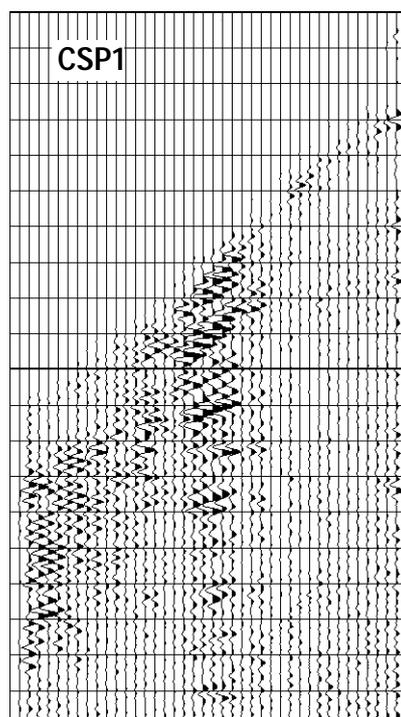
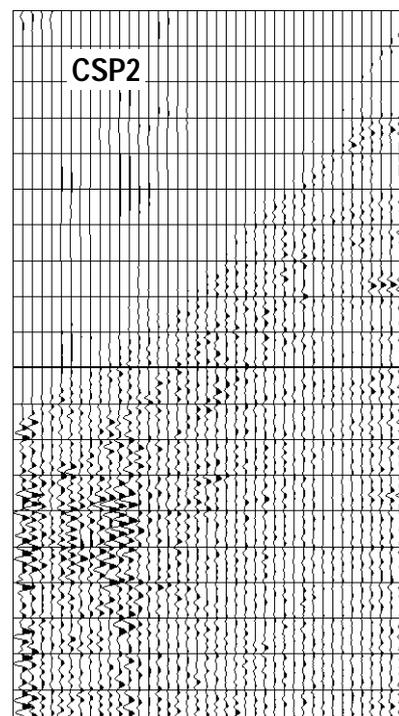
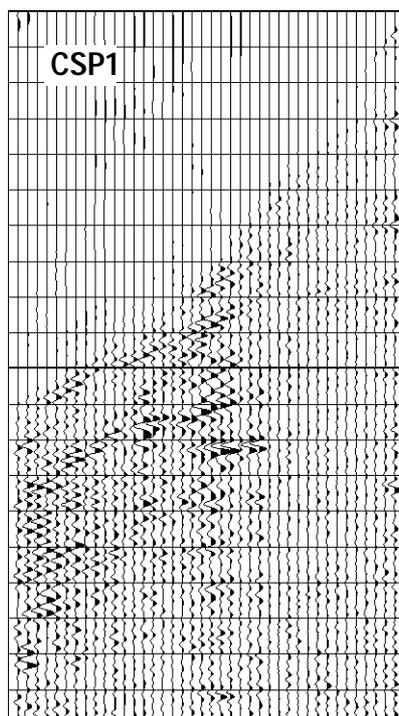


Figure 7.18 : Erreur sur régularisation de SI50 par LRD 3D en norme L1 (**haut**) et estimation des rapports S/B (**bas**).
vignettes 250x250m², recouvrement 100x100m²

SI50
R.R. norme L1
vign. 1000m
rec. 500m



SI50
R.R. S/B
vign. 1000m
rec. 500m

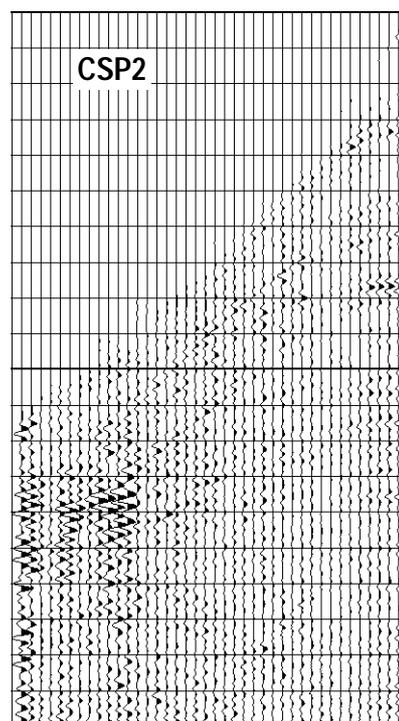
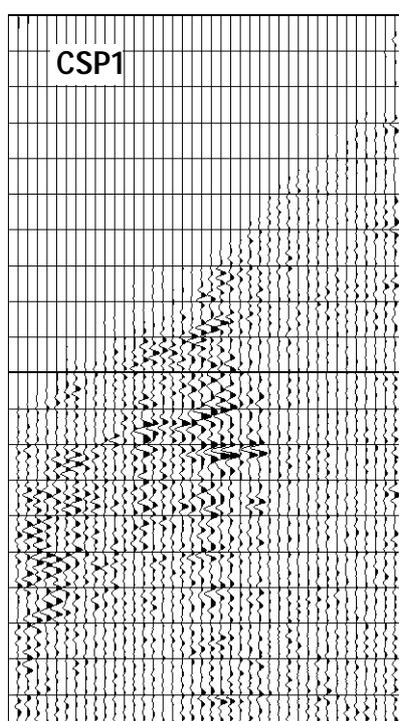


Figure 7.19 : Erreur sur régularisation de SI50 par LRD 3D en norme L1 (*haut*) et estimation des rapports S/B (*bas*).
vignettes 1000x1000m², recouvrement 500x500m²

Tableau 5 : pourcentage d'erreur sur régularisation de CSP2

LRD\vignette	250x250m ²	500x500m ²	1000x1000m ²
S/B	7.6%	4.0%	4.6%

7.4.6 - Interpolation (gaps moyens/grands)

Aucune des régularisations effectuées précédemment n'est capable d'interpoler correctement au travers du large gap au centre du point de tir CSP1 (figures 7.13, 7.13, 7.13 et 7.13). La taille de ce gap est 250x250m², c'est à dire 12 fois le pas d'échantillonnage théorique nécessaire (équ. (E7.3)), et 2 fois la distance d'interpolation en pratique accessible aux décompositions parcimonieuses (équ. (E7.4)).

Le résultat le plus acceptable est obtenu avec les décompositions parcimonieuses utilisées sur de grandes vignettes (1000x1000m²), encore faut-il préciser qu'il est n'acceptable que parce que les évènements dans cette zone sont relativement linéaires. Des décompositions plus parcimonieuses permettrait sans doute ici de mieux interpoler, mais les décompositions très parcimonieuses posent d'autres problèmes...

En définitive, un gap de taille allant jusqu'à $2R_v = 6\Delta x_{th}$ (équ. (E7.3) et (E7.4)) est interpolable avec les décompositions parcimonieuses utilisées ici. On constate d'ailleurs qu'en abaissant dans le critère de la figure 7.16 le rayon de 60m à 40m, les traces supplémentaires rejetées sont des traces correctement interpolées.

7.5 - Exemple sur données réelles: rééchantillonnage d'un PSV 3D

7.5.1 - Les données

Les données consistent en l'enregistrement d'un Profil Sismique Vertical (PSV) 3D (marine). Ce PSV a été acquis par cercles concentriques autour de la position des récepteurs dans le puits. L'outil est composé de 5 récepteurs à 3 composantes espacés de 12m. Il est placé à une profondeur verticale d'environ 2080m.

La distance nominale entre deux cercles est de 50m, l'inter-tir est de 25m. Sur les 38 cercles acquis, nous utilisons ici les 24 premiers, représentant 4770 tirs et allant jusqu'à un rayon d'environ 1350m (fig. 7.20). On note deux gaps importants dans cette acquisition: G1, situé au centre de l'acquisition, et G2, à cause de la présence de la plate-forme elle-même.

La fréquence maximum des données est d'environ 80Hz, avec une fréquence centrale de 30Hz (fig. 7.21). Les vitesses minimales relevées sont approximativement de 2500m/s, soit une lenteur de 0.4ms/m. La gamme de pentes possibles est donc $[p_{min}, p_{max}] = [-0.4, 0.4]$ ms/m. Le pas d'échantillonnage spatial devrait donc être $\frac{1}{f_{max}(p_{max} - p_{min})} \approx 15m$, ce qui signifie à priori que les données sont nettement aliasées (25m entre les tirs, 50m entre les cercles). En réalité, on trouve très peu d'énergie dans la gamme de fréquence au dessus de 60Hz, et dans les lenteurs plus grandes que 0.25ms/m (parce qu'on n'utilise pas les cercles les plus éloignés, qui vont jusqu'à un rayon de 2000m), ce qui réduit d'autant l'aliasing réel sur ces données.

L'objectif est de régulariser l'acquisition sur une grille rectangulaire $\Delta x = \Delta y = 25m$. On travaille sur la composante verticale du champ montant des ondes de compression (P), champ extrait par des techniques de filtrage matriciel (Gavin et al., 1994). Sur la figure 7.22 ce champs est tracé pour des

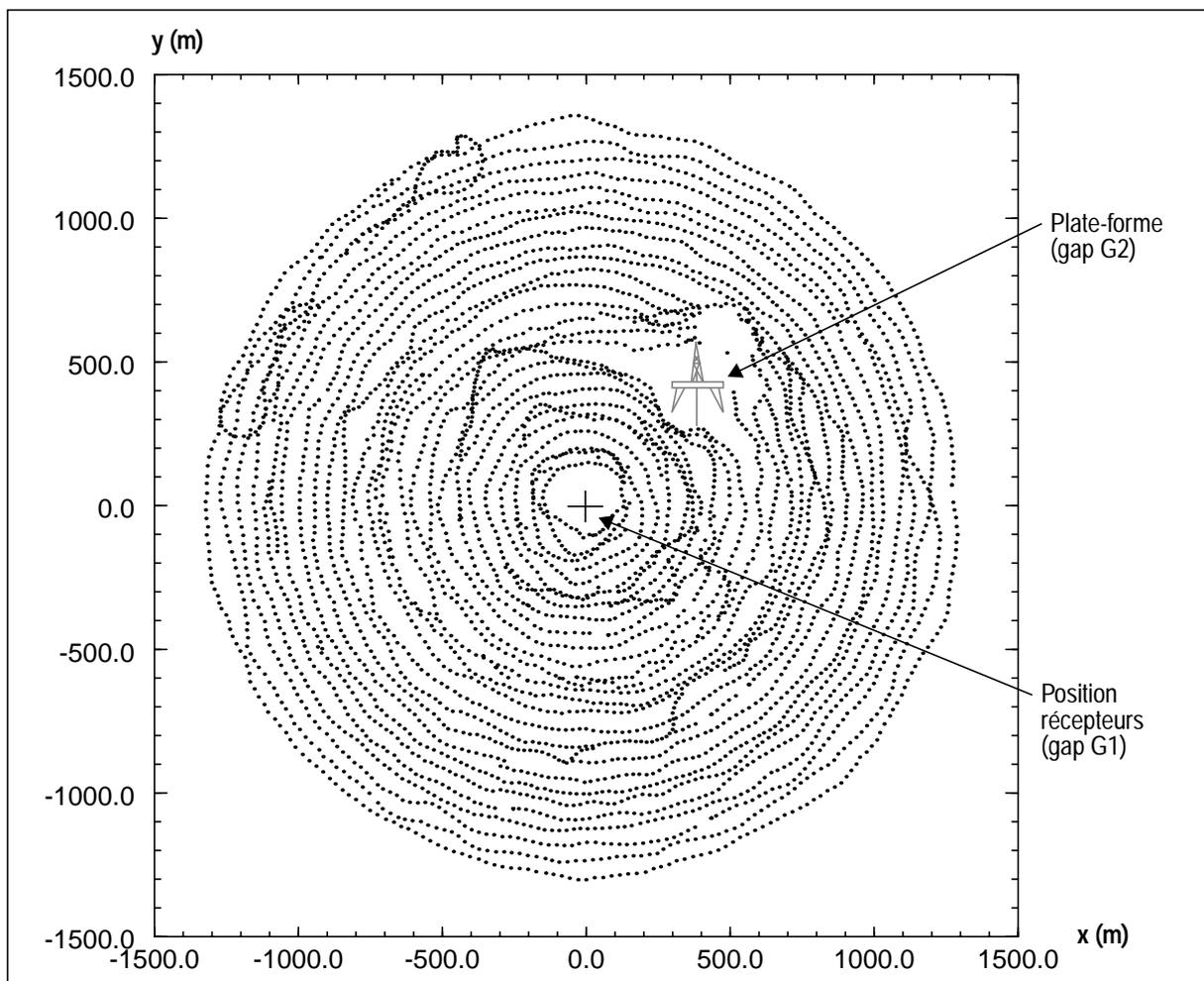


Figure 7.20 : position des points de tir du PSV 3D. L'acquisition est centrée sur la position des récepteurs (gap G1). Le grand gap vers $x=400m$, $y=500m$ (G2) est du à la présence de la plate-forme (il s'agit d'un puits dévié).

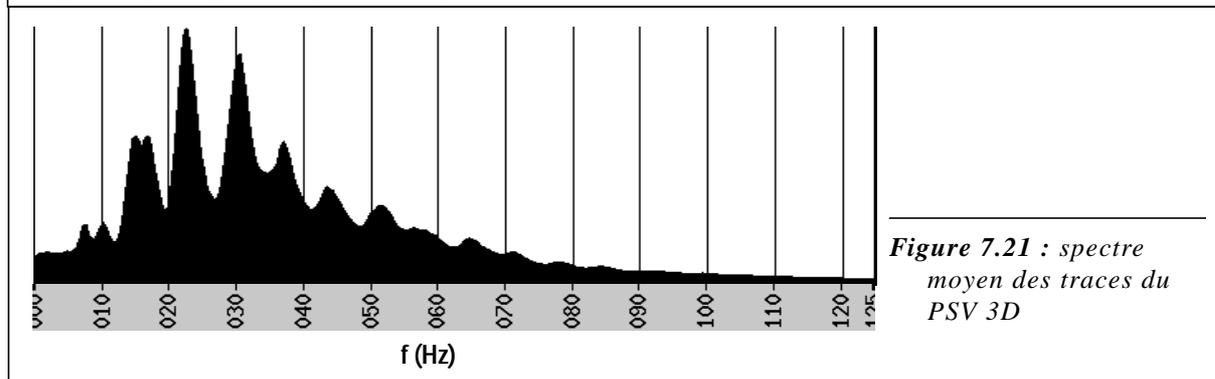


Figure 7.21 : spectre moyen des traces du PSV 3D

sections $y=cste$ (obtenues par simple binning $50 \times 50 m^2$) Une correction de divergence sphérique en $t^{1.5}$ a été appliquée.

7.5.2 - test de décimation

Dans un premier temps on effectue un test de décimation, qui consiste à supprimer une trace sur deux le long de chaque cercle (soit un tir sur deux en séquence), et à reconstruire ces traces. On compare ensuite les traces reconstruites aux traces enregistrées. Si le résultat est correct, on peut supposer que le résultat de la régularisation le sera également. On introduit également deux gaps (G1' et G2') de tailles comparables aux gaps déjà existant pour tester la capacité à interpoler ces derniers (fig. 7.23).

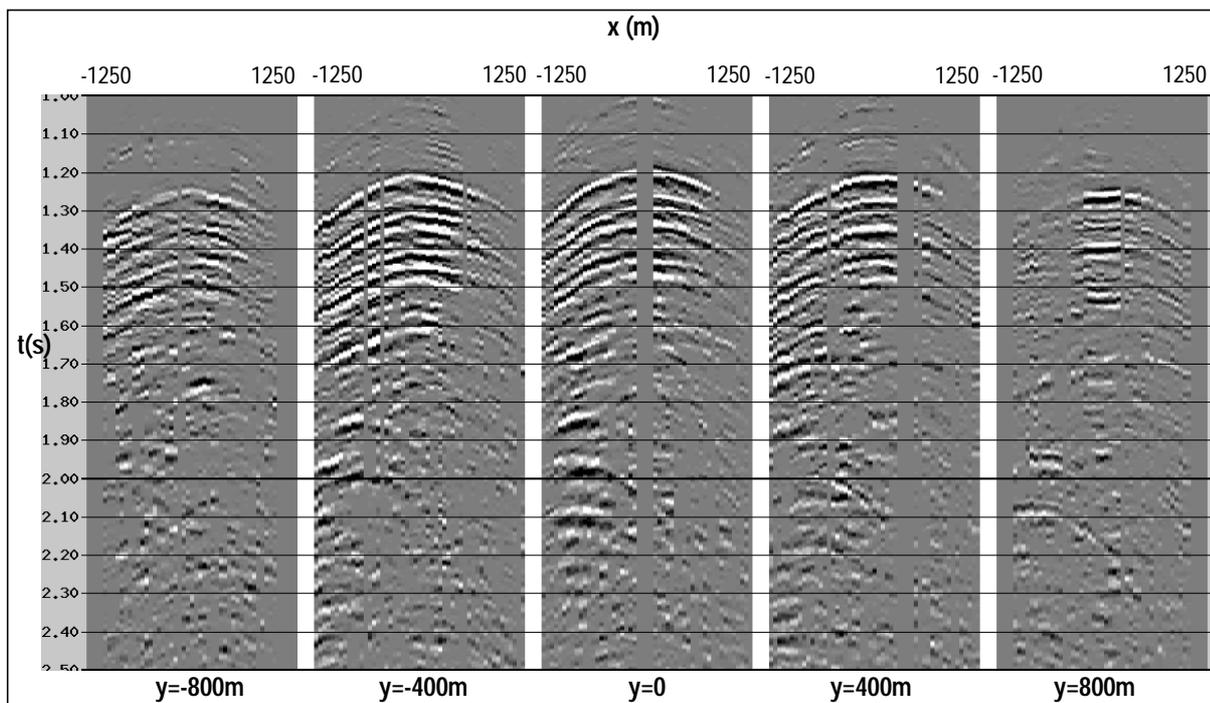


Figure 7.22 : Champs montant des ondes P extrait du PSV 3D (binning)

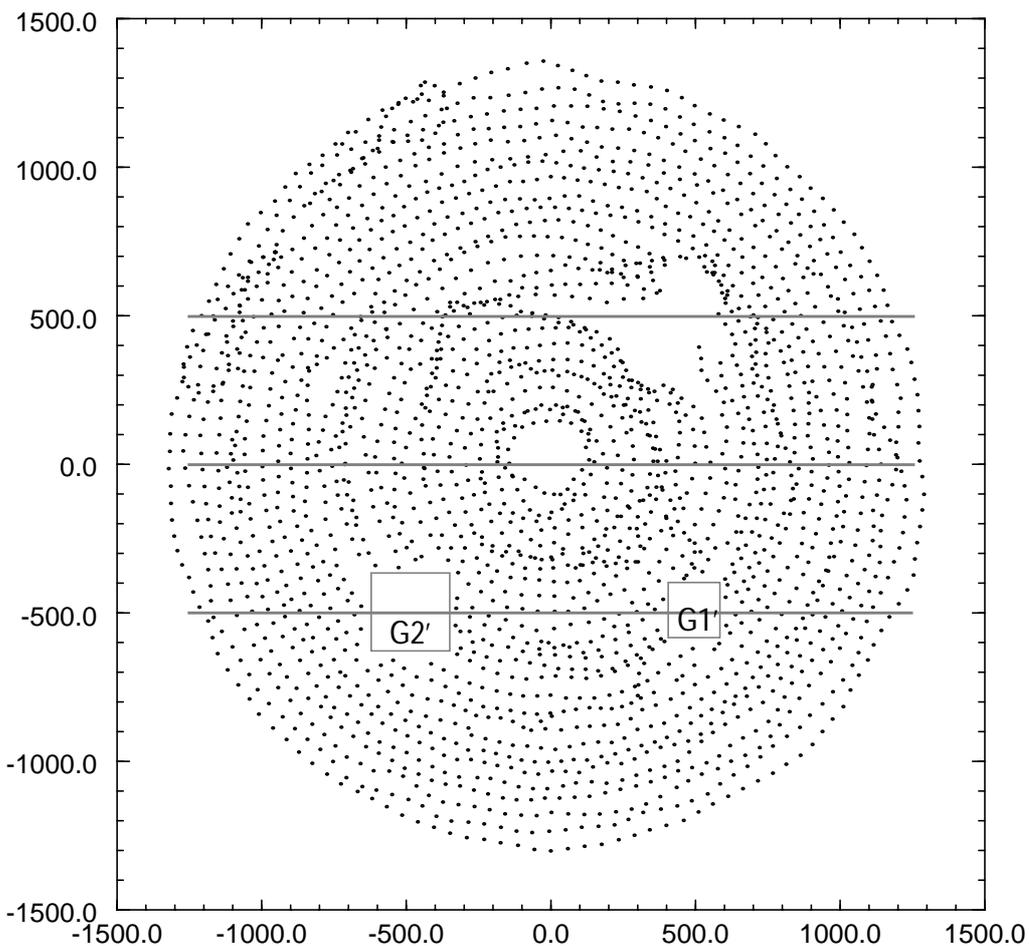


Figure 7.23 : décimation de l'acquisition fig. 7.20: 1 trace sur 2, gaps de $300 \times 300 \text{m}^2$ ($G2'$) et $200 \times 200 \text{m}^2$ ($G1'$) sur la section $y = -500 \text{m}$.

On effectue la reconstruction avec 3 types de LRD 3D (en norme L2, en norme L1, par estimation des rapports S/B), et deux tailles de vignettes (250x250m², recouvrement 150x150m², et 500x500m², recouvrement 250x250m²). Les pentes utilisées dans les LRD sont $-0.4 \leq p_x \leq 0.4$ ms/m et $-0.4 \leq p_y \leq 0.4$ ms/m. On visualise sur les figures 7.24 et 7.25 les erreurs sur 3 sections (y=-500, 0, 500m) obtenues par un binning élastique (bins de 25x25m). Sur les traces conservées dans la décimation l'erreur est bien sûr très faible.

On ne peut pas s'attendre ici à avoir des erreurs nulles, d'une part à cause du bruit aléatoire -par nature non prédictible-, et d'autre part parce la décimation a pu supprimer une partie de l'information. Une erreur importante ne signifie donc pas nécessairement que l'interpolation est mauvaise. Néanmoins on peut comparer ainsi plusieurs méthodes en comparant leurs erreurs.

On constate à nouveau que les LRD parcimonieuses donnent de meilleurs résultats que la LRD en norme L2. L'amélioration n'est toutefois pas aussi spectaculaire que dans les tests sur synthétiques, sans doute en partie parce que les données n'étant pas très aliasées, le résultat obtenu avec la norme L2 est déjà relativement bon. Les gaps G1' et G2' ne sont pas correctement interpolés, même avec les LRD parcimonieuses.

Autre différence avec le cas synthétique, les résultats obtenus pour les décompositions parcimonieuses avec la petite vignette (250m) semblent meilleurs qu'avec la grande (500m). Les données ne sont en fait pas du tout semblables: on note en effet beaucoup de discontinuités dues à la géologie, alors que les données synthétiques étaient très lisses (car calculées à partir d'un modèle lisse également). Ces discontinuités contredisent bien sûr l'hypothèse des événements plans d'amplitudes constantes, et statistiquement une petite vignette contient moins de discontinuités qu'une grande, d'où le meilleur résultat.

Le tableau suivant donne l'énergie résiduelle (en pourcentage de l'énergie des données de départ) de l'erreur pour chaque test. Il confirme le constat visuel sur la taille de la vignette:

Tableau 6 : énergie de l'erreur (en %) des reconstructions après décimation.

LRD\vignette	250x250m ²	500x500m ²
norme L2	19.0%	18.7%
norme L1	12.4%	14.9%
estimations S/B	12.3%	12.9%

L'amélioration apportée par les décompositions parcimonieuses n'est pas très importante. Elle est plus sensible si on ne considère que la partie la plus haute fréquence des données (la plus aliasée). On applique sur tous les résultats un filtre passe-haut (avec une rampe entre 30Hz et 50Hz):

Tableau 7 : énergie de l'erreur (en %) des reconstructions après décimation. Partie HF des données.

LRD\vignette	250x250m ²	500x500m ²
norme L2	49.6%	48.2%
norme L1	26.6%	34.2%
estimations S/B	24.7%	27.9%

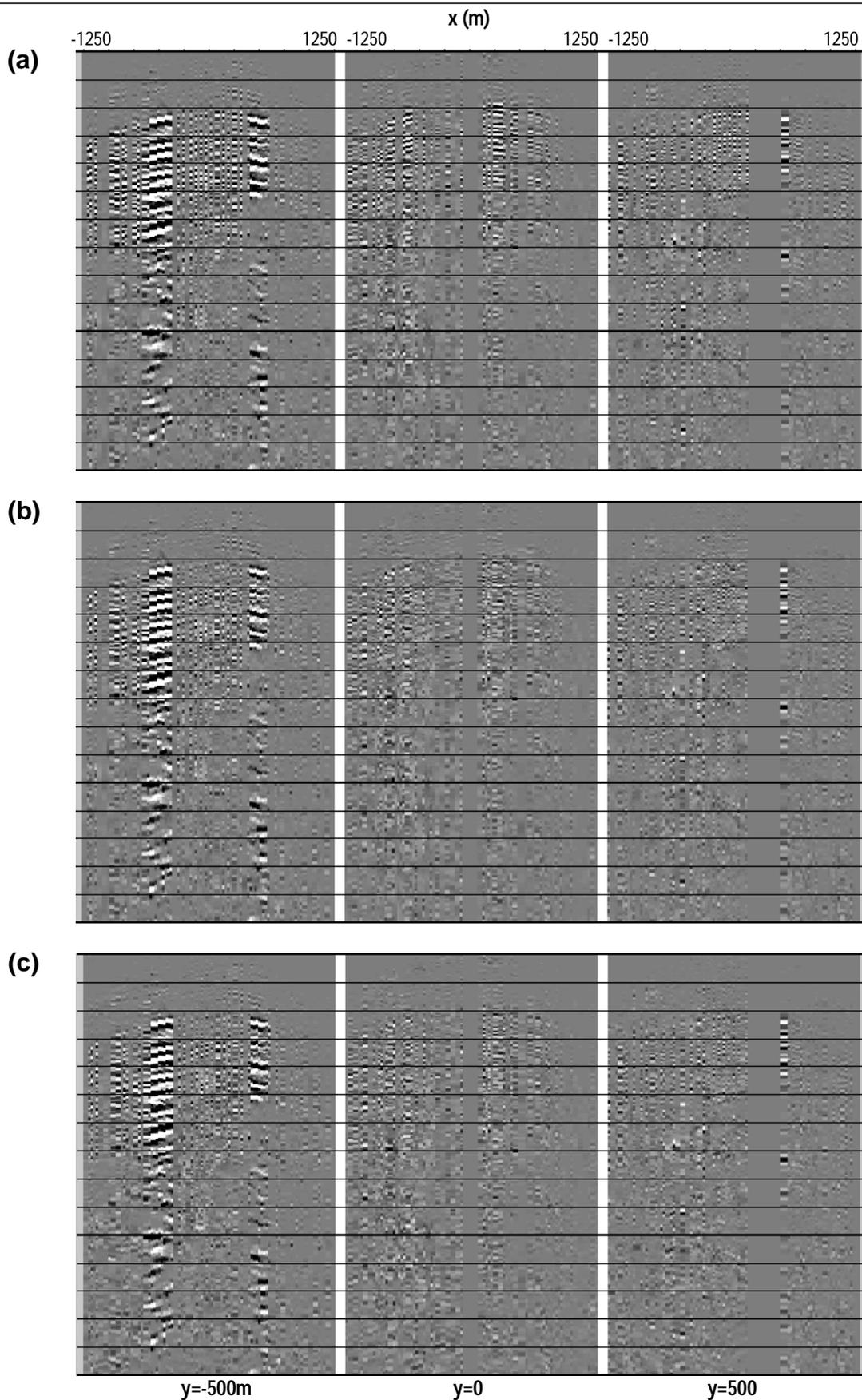
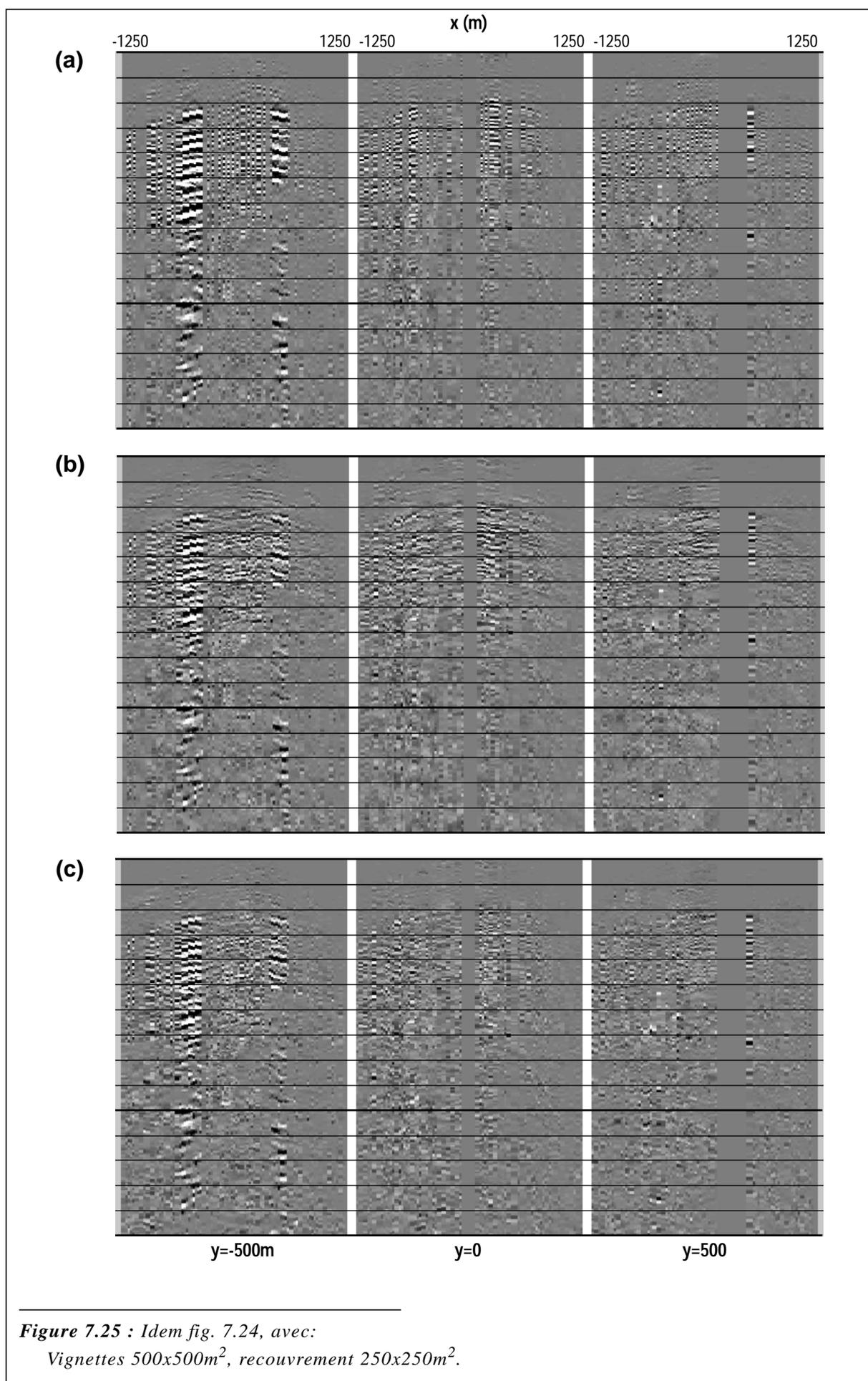


Figure 7.24 : erreurs de la reconstruction de l'acquisition originale fig. 7.20 à partir de la décimation fig. 7.23; visualisation de 3 sections $y=-500, 0, 500\text{m}$, obtenues par binning élastique. LRD 3D (a) en norme L2, (b) en norme L1, (c) estimation S/B. Vignettes $250 \times 250\text{m}^2$, recouvrement $150 \times 150\text{m}^2$.



Les décompositions parcimonieuses permettent de diviser l'erreur par 2 sur la partie la plus HF des données. Dans l'absolu, les forts pourcentages relevés dans ce cas s'expliquent sans doute par un mauvais rapport S/B dans les HF (le signal étant faible dans ces fréquences).

Pour conclure ce test de décimation, les LRD 3D parcimonieuses permettent de reconstruire les données décimées avec une qualité satisfaisante, sauf pour l'interpolation à travers les deux gaps introduits. La régularisation de l'acquisition originale est donc envisageable, mais il ne faut à priori pas espérer combler correctement les deux gaps présents dans les données elles-mêmes.

7.5.3 - régularisation

On veut régulariser l'acquisition du PSV 3D sur une grille rectangulaire où $\Delta x = \Delta y = 25\text{m}$ (fig. 7.26). On visualise les résultats sur trois sections ($y=0, y=400\text{m}, y=800\text{m}$), dont les deux premières traversent les gaps G1 et G2.

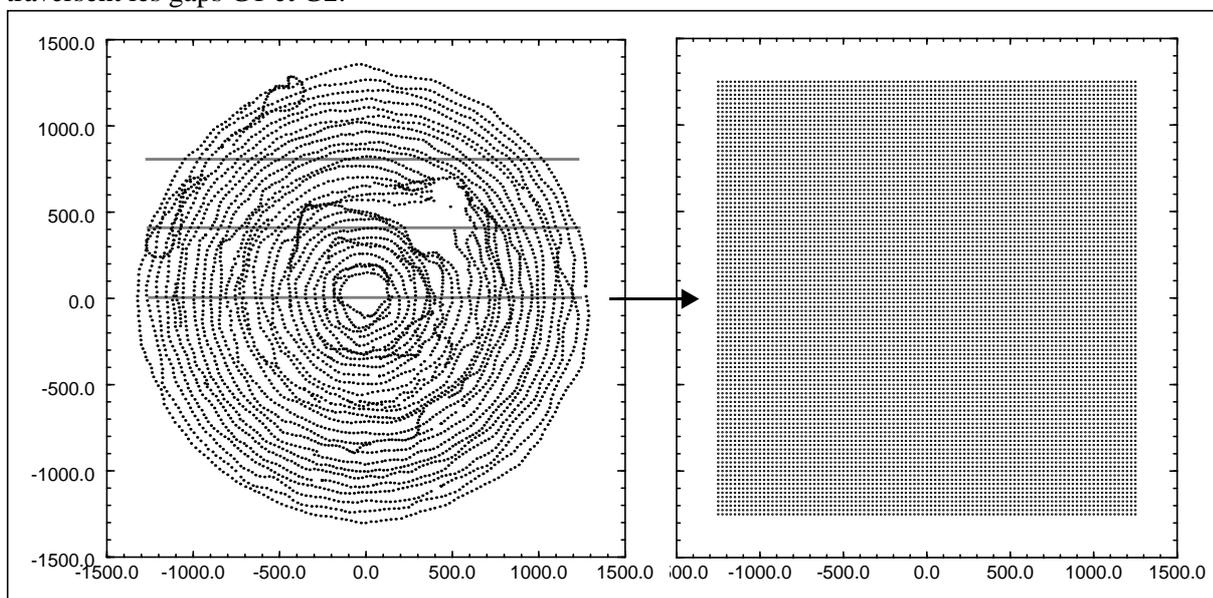


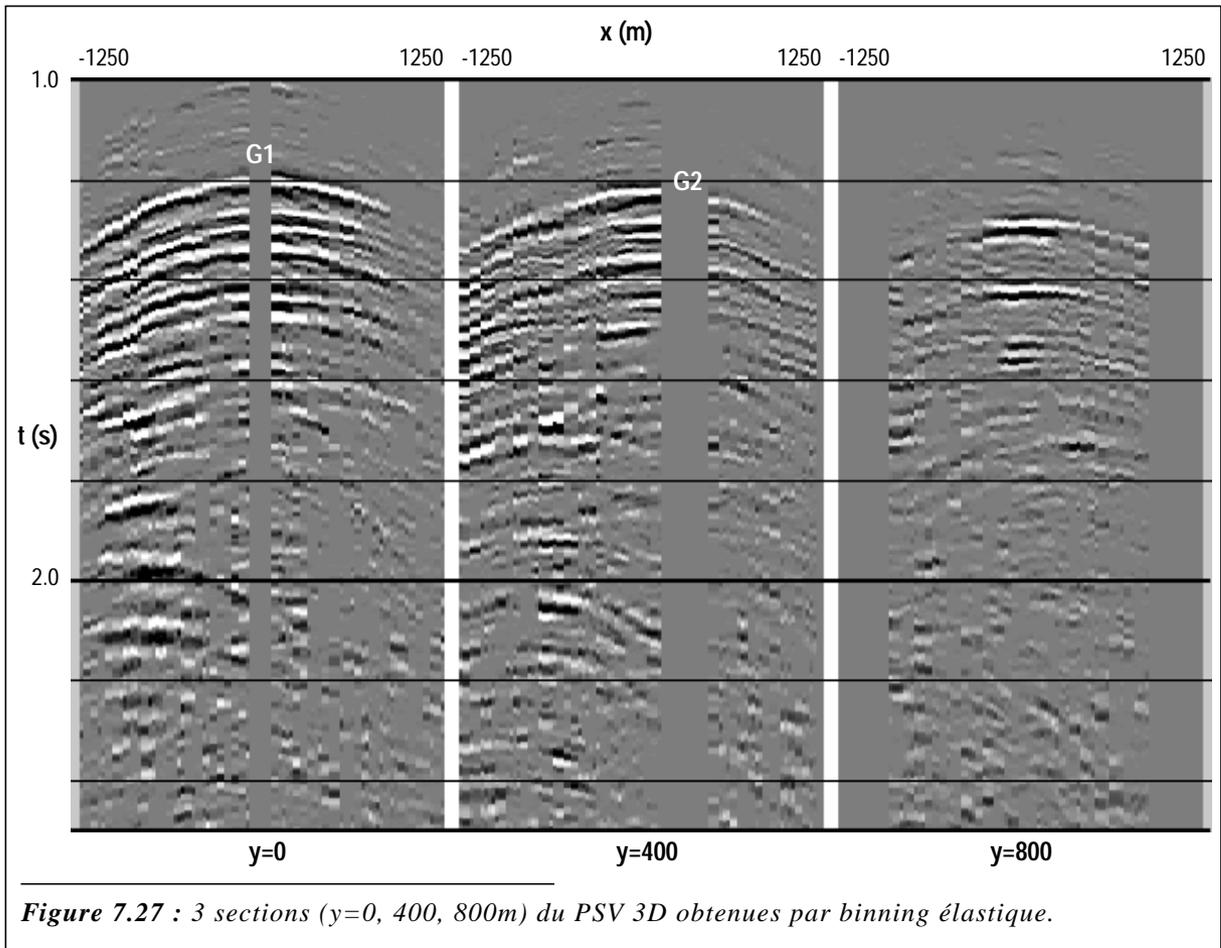
Figure 7.26 : régularisation de l'acquisition du PSV 3D sur une grille rectangulaire 25x25m

Nous testons d'abord un binning élastique (fig. 7.27) pour visualiser rapidement les trois sections, et pour vérifier que ce n'est la solution à notre problème: il y a de nombreux bins vides, ce qui entraîne la duplication de nombreuses traces. D'une manière générale les évènements pentés ne sont plus en phase.

Les autres tests consistent en des régularisations par LRD 3D (RR signifie Régularisation de Radon):

- **RR1:** LRD en norme L2, vignettes $250 \times 250\text{m}^2$, recouvrement $150 \times 150\text{m}^2$ (fig. 7.28a)
- **RR2:** LRD en norme L1, vignettes $250 \times 250\text{m}^2$, recouvrement $150 \times 150\text{m}^2$ (fig. 7.28b)
- **RR3:** LRD en norme L1, vignettes $500 \times 500\text{m}^2$, recouvrement $250 \times 250\text{m}^2$ (fig. 7.28c)
- **RR4:** LRD par estimations S/B, vignettes $500 \times 500\text{m}^2$, recouvrement $250 \times 250\text{m}^2$ (fig. 7.28d)

Le résultat RR1 semble déjà très correct, bien que les deux gaps G1 et G2 ne soient pas comblés. RR2, par rapport à RR1, permet de combler -au moins en partie- G1, mais pas G2. Pour le reste, les différences entre les deux tests ne sont pas évidentes à l'oeil nu. RR3, en utilisant de plus grandes vignettes, permet de combler le G2, mais très imparfaitement (au niveau des amplitudes notamment). RR4 n'améliore pas notablement le résultat de RR3.

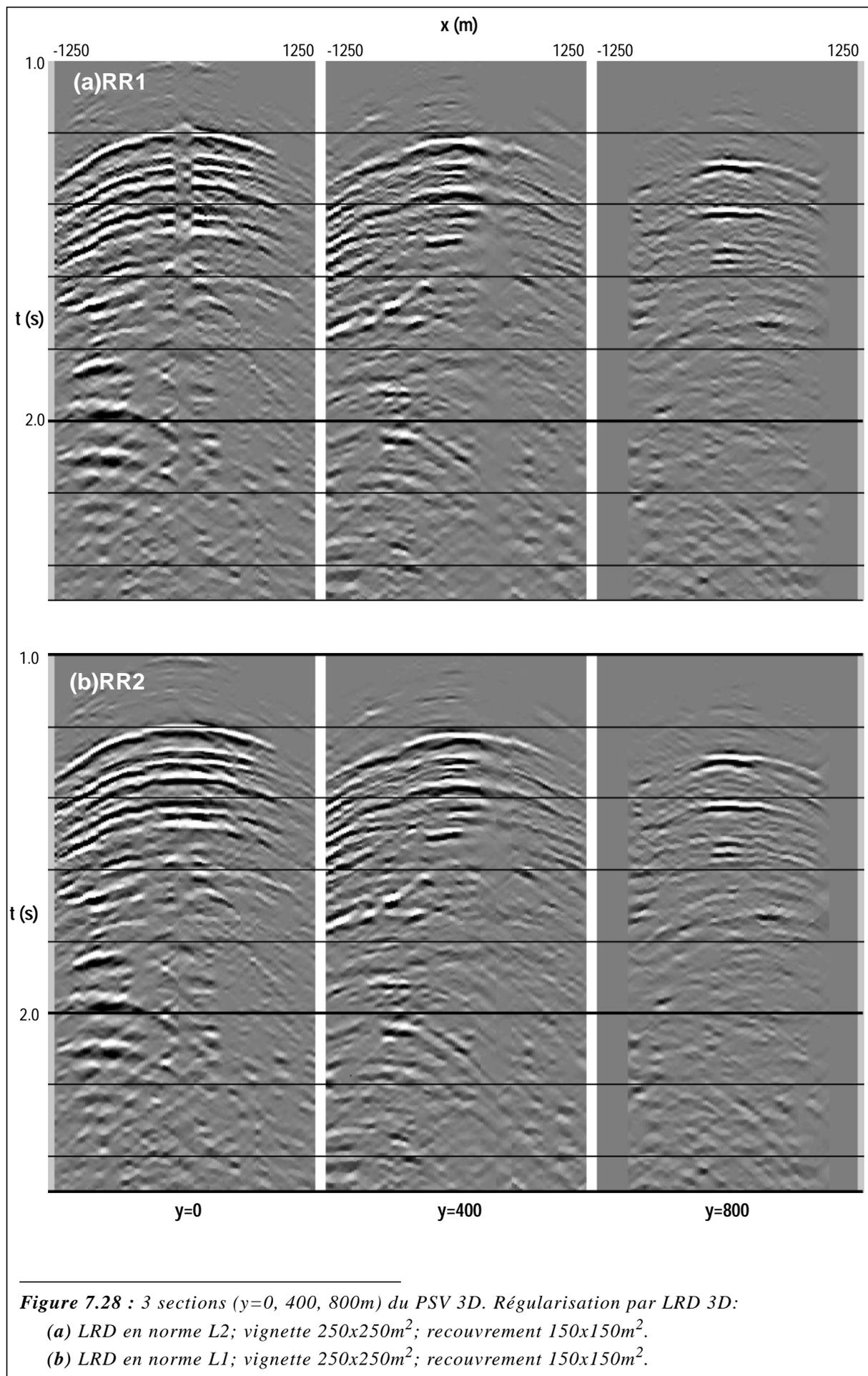


On peut s'étonner de parvenir à combler G1, alors que dans les tests de décimations nous n'étions pas parvenu à combler G1', pourtant de taille comparable. A cela il y a deux explications: d'une part la gamme de pentes utilisé ici est plus réduite que pour les tests de décimation (pentes maximum de 0.25ms/m au lieu de 0.4ms/m), réduisant donc l'aliasing dans la décomposition, et corollairement augmentant la portée de l'interpolation; d'autre part les événements les mieux interpolés au niveau de G1 sont sub-horizontaux (G1 est au centre de l'acquisition) et sont donc moins susceptibles d'être aliasés que des événements pentés (que l'on observe aussi vers $t=2.0\text{s}$ au niveau de G1, et qui effectivement sont moins bien interpolés).

De RR1 à RR4 nous avons donc utilisé des LRD de plus en plus parcimonieuses, et des vignettes de plus en plus grandes. En conséquence de quoi l'interpolation est possible à travers des gaps de plus en plus larges, et le résultat obtenu est de plus en plus lisse. RR4 est en effet extrêmement lisse, et ne correspond sans doute pas à la réalité. De ce point de vue RR1 est plus réaliste, même si de nombreuses traces interpolées ne sont sans doute pas valides suivant le critère de la figure 7.16.

Avec une certaine part de subjectivité, le résultat le plus intéressant est ici RR2: il n'est pas trop lisse (petite vignette), et la LRD en norme L1 permet d'interpoler la plupart des traces.

Une autre solution serait de combiner ces différents résultats: de RR1 on peut conserver les traces valides suivant le critère figure 7.16 avec $R_v=20\text{m}$, et pour les autres traces prendre le résultat de RR2 (ou RR3, ou RR4). Il est possible de raffiner un peu plus en sélectionnant les basses fréquences (moins aliasées) sur un résultat et les hautes fréquences sur un autre. En effet si pour des données de fréquence maximum f_{max} le rayon d'interpolation possible avec la LRD en norme L2 est R_v , alors pour les mêmes données filtrées dont on ne garde que les BF jusqu'à $f_{\text{max}}/2$ le rayon d'interpolation possible est $2 R_v$. On peut affirmer cela parce que la LRD en norme L2 est linéaire par rapport aux données, donc que la LRD de la partie BF des données est égale à la partie BF de la LRD des données (ce n'est pas le cas avec les LRD parcimonieuses).



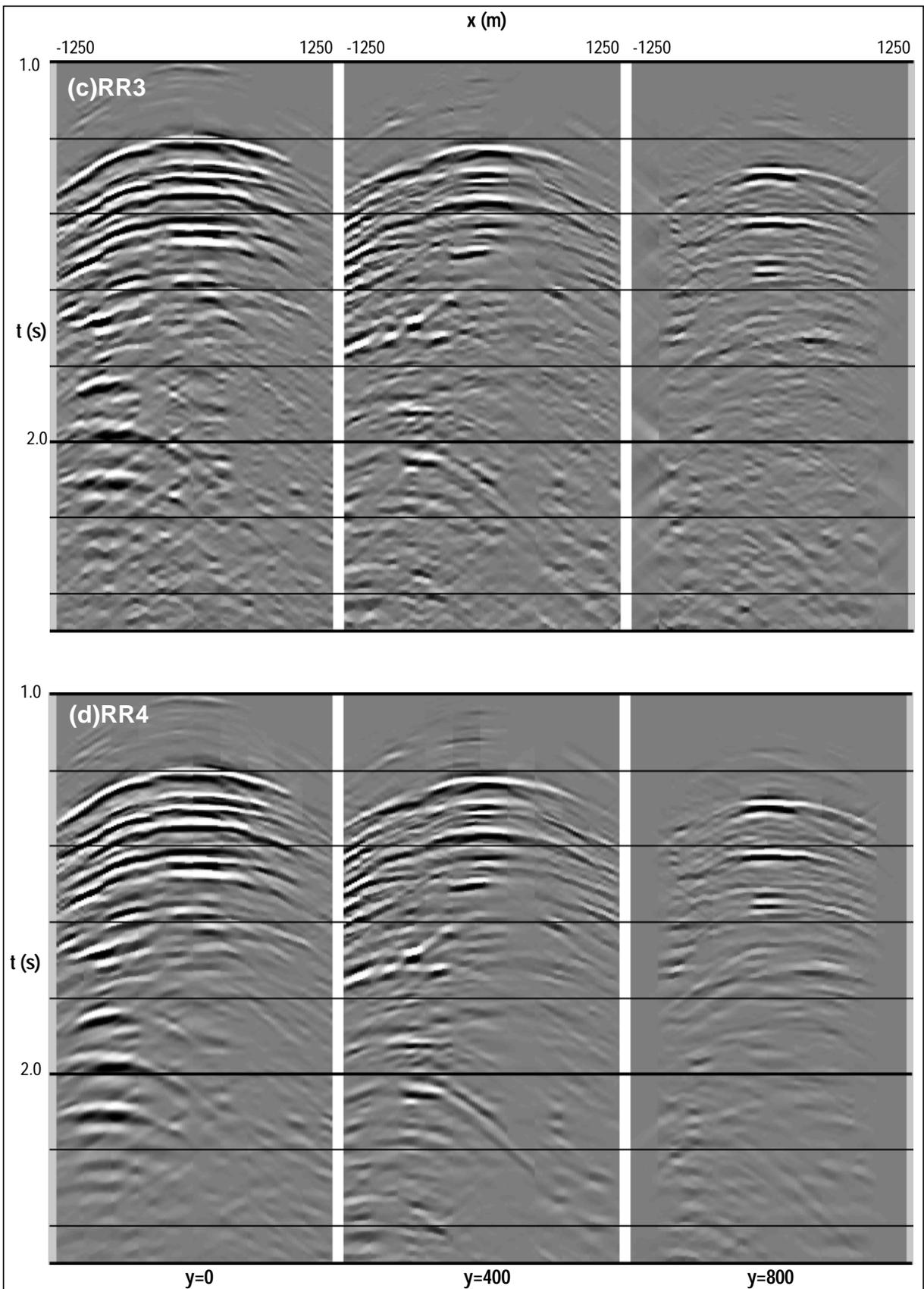


Figure 7.28 (suite): 3 sections ($y=0, 400, 800\text{m}$) du PSV 3D. Régularisation par LRD 3D:
 (c) LRD en norme L1; vignette $500 \times 500\text{m}^2$; recouvrement $250 \times 250\text{m}^2$.
 (d) LRD par estimation S/B; vignette $500 \times 500\text{m}^2$; recouvrement $250 \times 250\text{m}^2$.

Ainsi le résultat de la figure 7.28 est obtenu en combinant RR1 et RR3: si la trace vérifie le critère pour $R_v=20m$, on utilise le résultat de RR1; si elle vérifie le critère pour $R_v=40m$, on utilise la partie (0-30Hz) de RR1 et la partie (30Hz- f_{max}) de RR3 (on filtre en fait avec une rampe entre 20 et 40Hz); sinon on utilise le résultat de RR3. On conserve donc le caractère réaliste de RR1, en profitant de la meilleure interpolation obtenue avec RR3.

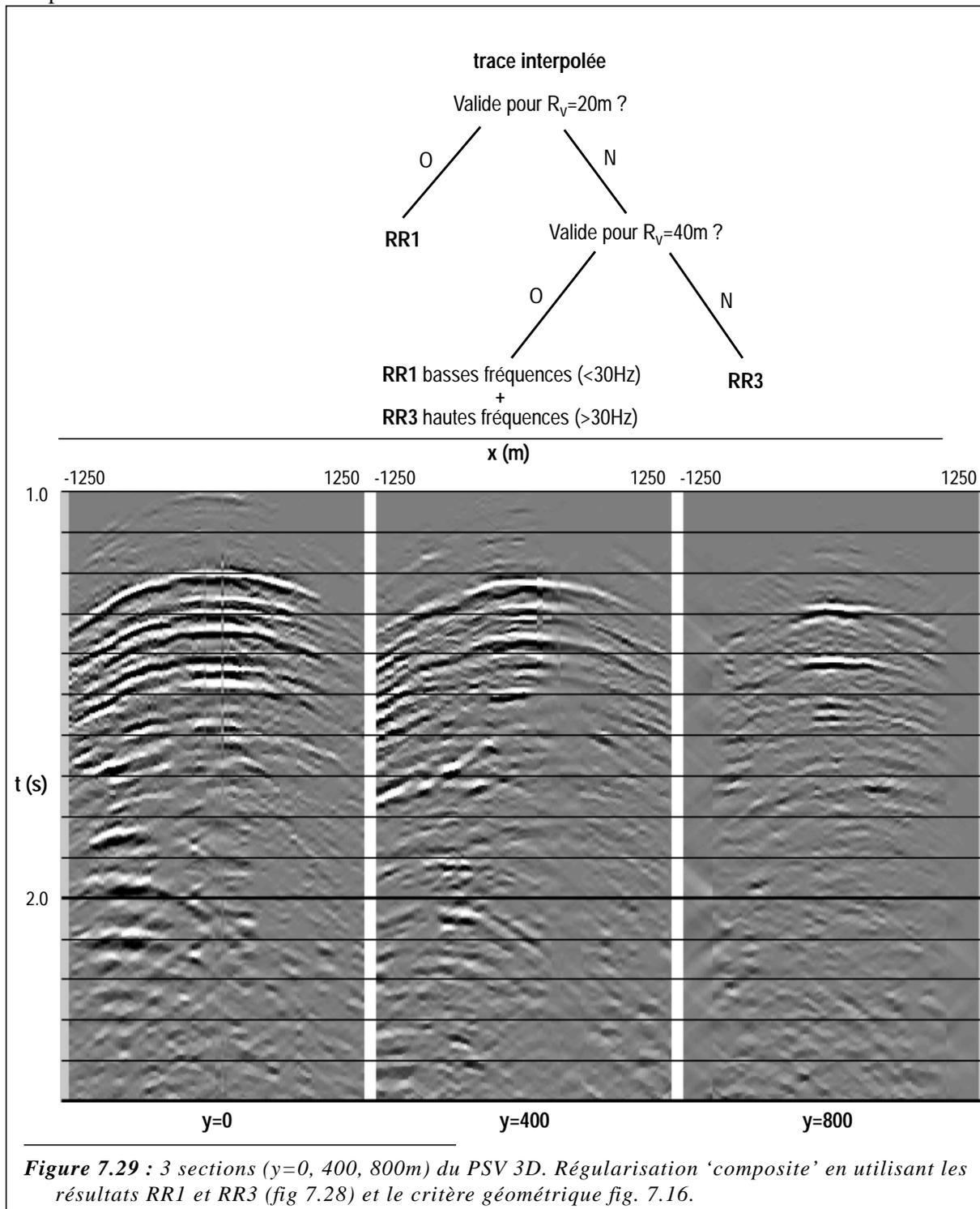


Figure 7.29 : 3 sections ($y=0, 400, 800m$) du PSV 3D. Régularisation 'composite' en utilisant les résultats RR1 et RR3 (fig 7.28) et le critère géométrique fig. 7.16.

Dans tous les cas, les grands gaps semblent difficiles à combler par ces techniques de décompositions de Radon.

7.5.4 - Amélioration de la qualité après rééchantillonnage

Afin de juger de l'amélioration que peut apporter un rééchantillonnage sur toute la chaîne de traitement en aval, nous avons calculé des décompositions globales en ondes planes (PWD, pour plane-wave decomposition) du PSV. Par global on entend en utilisant toutes les données d'un seul bloc, sans utiliser de vignettes. Il s'agit en fait d'une LRD 3D en norme L2 (dans la PWD on peut par exemple effectuer la déconvolution des ondes montantes par les descendantes, mais ce n'est pas le sujet ici).

Nous avons calculé deux PWD (fig 7.30): la première à partir des données irrégulières d'origine, la seconde après un rééchantillonnage par LRD 3D en norme L1 sur une grille régulière rectangulaire 25x25m (vignettes de 250x250m²). Il apparaît clairement que la seconde est beaucoup moins bruitée que la première, en grande partie parce que l'aliasing spatial a été réduit dans l'opération de rééchantillonnage.

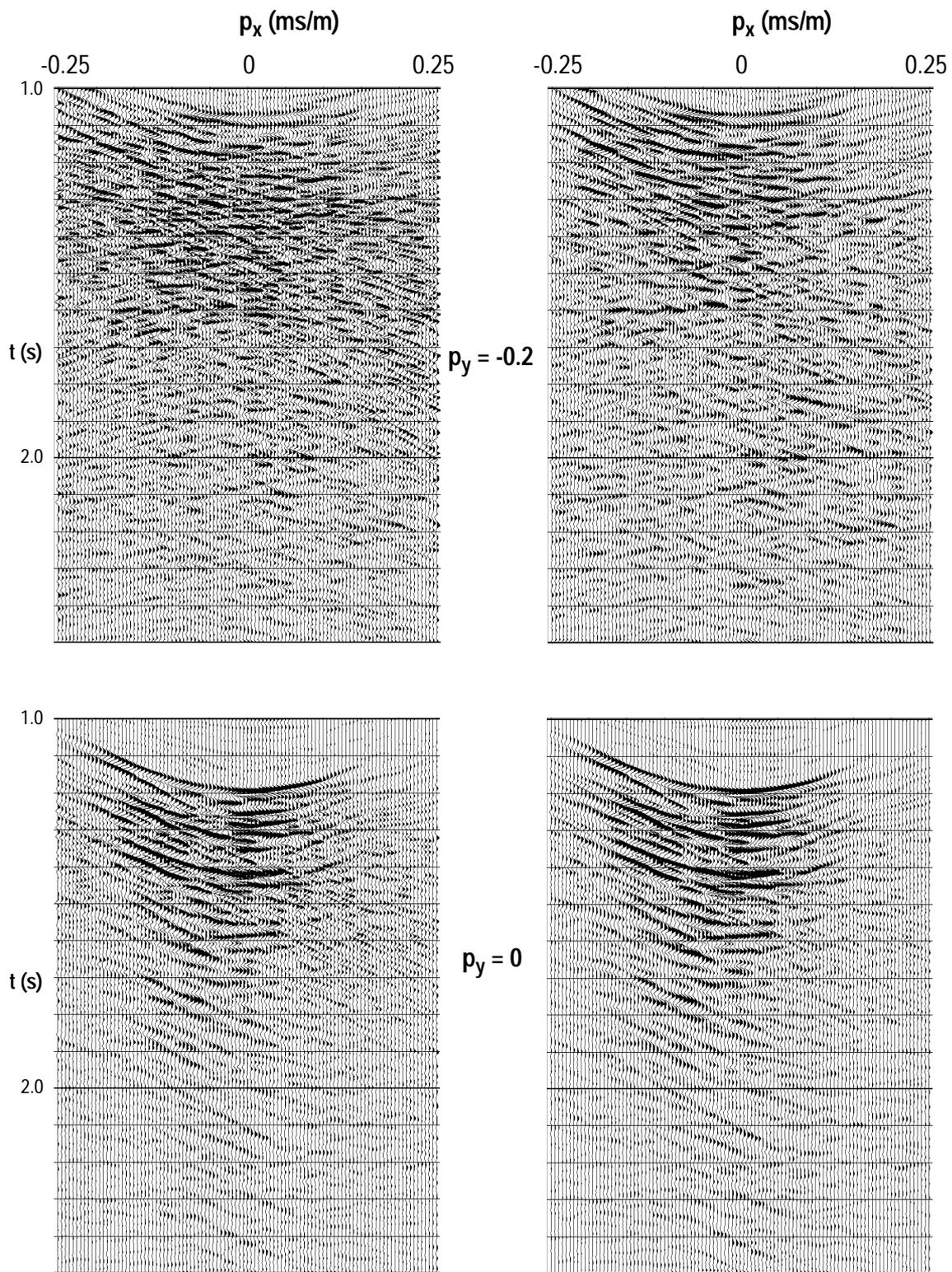


Figure 7.30 : décomposition globale en plane du PSV 3D. (*gauche*) à partir des données d'origine; (*droite*) à partir des données rééchantillonnées par décompositions de Radon en norme L1 (vignettes 250x250m²). (deux sections $p_y=cste$ tracées).

7.6 - Conclusions

Nous avons vu dans ce chapitre que les décompositions de Radon permettent de rééchantillonner et d'interpoler spatialement des données suivant 2 directions spatiales.

Les décompositions en norme L2 ne permettent pas en principe de traiter les données aliasées, ce qui limite leur application au rééchantillonnage des données non aliasées.

Les décompositions parcimonieuses utilisées ici permettent d'interpoler au travers de gaps jusqu'à 6 fois (valeur déterminée empiriquement) le pas d'échantillonnage théoriquement nécessaire. Les gaps plus importants ne sont pas accessibles aux méthodes de Radon.

La méthodologie utilisée découpe les données en vignettes de taille prédéfinie que l'on traite séparément. Les décompositions en norme L2 sont peu sensibles à la taille choisie, mais les décompositions parcimonieuses fonctionnent mieux sur des petites vignettes (les petites vignettes sont en outre préférables pour le temps de calcul), et ce d'autant plus si les données ne sont pas lisses. D'une manière générale, les résultats obtenus sont d'autant plus lisses que la décomposition utilisée est parcimonieuse et que la taille des vignettes est importante.

Le problème du temps de calcul important des décompositions parcimonieuses, argument souvent avancé, doit être relativisé: sur des données aliasées on ne peut tout simplement pas utiliser les décompositions en moindres carrés, on n'a donc pas le choix. Les méthodes utilisées ici représentent un surcoût pour les décompositions parcimonieuses de l'ordre de 5 fois. Mais il ne faut pas oublier que l'on peut ainsi traiter des données avec un échantillonnage 3 fois plus faible environ, ce qui correspond à 9 fois moins de traces pour un problème 3D, et que le temps de calcul est directement proportionnel au nombre de traces dans les données (une fois fixés tous les autres paramètres).

Conclusions... et plus si affinités

Après avoir bien posé le problème pour les décompositions de Radon 2D linéaires, paraboliques, et hyperboliques, nous avons pu étudier et bien comprendre les problèmes d'échantillonnage et d'aliasing. L'aliasing spatial, sur lequel on ne peut pas revenir une fois les données acquises -à moins d'éliminer volontairement les parties aliasées- a pour conséquence une dispersion de l'énergie des plus hautes fréquences (temporelles) dans le domaine transformé, sans pour autant empêcher une bonne décomposition, au sens que les résidus sont faibles.

Les décompositions en moindres carrés, outre qu'elles ne traitent pas correctement les données aliasées et qu'elles focalisent mal les événements, apparaissent peu sensibles à l'estimation du niveau de bruit (mais en contrepartie elles ne permettent pas de le filtrer). Pour la décomposition de Radon linéaire on peut construire un filtre optimum de préconditionnement de type 'filtre rho', adapté à la géométrie globale des données. Combiné à une pondération destinée à compenser l'irrégularité éventuelle des données, il permet d'obtenir une décomposition en moindres carrés en moins de cinq itérations de gradient conjugué pour toute donnée.

Les décompositions parcimonieuses, qui consistent à expliquer les données avec le moins possible d'échantillons non nuls du modèle, constituent une réponse adaptée au problème de l'aliasing spatial. Elles sont effectuées par une approche de type IRLS (Iteratively Reweighted Least Squares), permettant d'obtenir des implémentations par gradient conjugué proche directement dérivées de celle des décompositions en moindres carrés. En étudiant la convergence de ces algorithmes, il est possible de limiter le surcoût -inévitables- par rapport aux décompositions en moindres carrés à des valeurs raisonnables (de l'ordre de 5 fois). Ces décompositions restent néanmoins délicates à mettre en oeuvre de part leur sensibilité au niveau de bruit et les conséquences possibles (perte des événements peu énergiques) d'un mauvais réglage des paramètres associés (en contrepartie le bruit aléatoire peut être filtré efficacement). Une estimation automatique du signal et du bruit le long de chaque courbe de la famille utilisée résout ce problème.

La décomposition de Radon linéaire 3D (2 variables spatiales) est une extension relativement triviale de la décomposition 2D (sauf dans le domaine de Fourier où le problème se décompose en 2 fois 2D ou non suivant que les données sont disposées sur une grille régulière ou non, mais toutes nos implémentations sont en domaine temps), à l'exception des problèmes de convergence qui rendent l'utilisation de filtres rho optimums (dérivés du cas 2D) indispensable.

L'interpolation et le rééchantillonnage de traces suivant deux directions spatiales en utilisant les décompositions de Radon linéaires 3D montrent que les décompositions parcimonieuses permettent d'interpoler et des traces au travers de gaps dont le taille va jusqu'à 6 fois le pas d'échantillonnage théorique (alors que les décompositions en moindres carrés ne peuvent interpoler des données aliasées). Le respect des hypothèses de parcimonie, ainsi que des considérations de temps de calcul, imposent de travailler sur des fenêtres spatiales glissantes de petites dimensions (d'autant plus que les données ne sont pas lisses): typiquement de l'ordre de 250 mètres. Par ce procédé on peut donc réduire le pas d'échantillonnage d'une grille régulière, reconstruire des traces manquantes dans des gaps de tailles moyennes, ou rééchantillonner complètement une acquisition. L'interpolation au travers de grands gaps nécessite d'autres techniques.

Les directions de recherches futures peuvent concerner les décompositions sur des familles de courbes dont les profils d'amplitudes ne sont pas fixés mais paramétrés. L'application principale étant la prise en compte des effets AVO dans les collections CMP. Les résultats obtenus ici sur des données réelles sont décevants, à cause de la forte sous-détermination du problème, même avec des décompositions parcimonieuses. Il faudrait contraindre plus fortement la décomposition, en n'autorisant par exemple la paramétrisation qu'en un nombre réduit d'échantillons du modèle, obtenus par un pointé préalable des évènements.

Il serait également intéressant de tester des décompositions de Radon linéaires 4D, voire 5D (4 variables spatiales). En effet une acquisition sismique de surface caractérise chaque trace par 4 variables spatiales (abscisses et ordonnées de la source et du récepteur). Il ne devrait en effet y avoir aucun obstacle théorique, seul le temps de calcul pouvant être éventuellement rédhibitoire.

Annexe A: le Gradient Conjugué

A.1 - Introduction

Le gradient conjugué est une méthode d'optimisation itérative qui s'applique généralement à des problèmes comportant un grand nombre d'inconnues que l'on ne peut résoudre par des méthodes directes. Il est défini initialement pour des fonctions quadratiques, mais il peut être appliqué à des cas plus généraux (soit que l'on puisse les approximer par des fonctions quadratiques, soit que l'on sache calculer -ou simplement approximer- le gradient).

Soit φ une forme quadratique de \mathbb{R}^n dans \mathbb{R} :

$$\begin{aligned} \mathbb{R}^n &\rightarrow \mathbb{R} \\ \mathbf{x} &\rightarrow \varphi(\mathbf{x}) = \varphi(\mathbf{0}) - \mathbf{b} \cdot \mathbf{x} + \frac{1}{2} \mathbf{x} \cdot \mathbf{A} \mathbf{x} \end{aligned} \tag{EA.1}$$

où \mathbf{A} est une matrice carrée ($n \times n$) définie, positive, symétrique, et \mathbf{b} un vecteur de \mathbb{R}^n .

Les dérivées premières (gradient) et secondes (hessien) de φ sont ici connues:

$$\begin{aligned} \nabla \varphi(\mathbf{x}) &= -\mathbf{b} + \mathbf{A} \mathbf{x} \\ \nabla^2 \varphi(\mathbf{x}) &= \mathbf{A} \end{aligned} \tag{EA.2}$$

On cherche le minimum de $\varphi(\mathbf{x})$ par rapport à \mathbf{x} : cela est équivalent à vouloir résoudre le système linéaire $\mathbf{Ax}=\mathbf{b}$ (au minimum, le gradient de φ doit être nul).

$$\min_{\mathbf{x} \in \mathbb{R}^n} \left(\varphi(\mathbf{x}) = \left(\varphi(\mathbf{0}) - \mathbf{b} \cdot \mathbf{x} + \frac{1}{2} \mathbf{x} \cdot \mathbf{Ax} \right) \right) \quad (EA.3)$$

$$\Downarrow$$

$$\mathbf{Ax} = \mathbf{b}$$

Posons $\mathbf{r} = \mathbf{b} - \mathbf{Ax}$. Il s'agit du résidu lié au système linéaire $\mathbf{Ax}=\mathbf{b}$. Il est égal, au signe près, au gradient de φ : $\nabla\varphi = -\mathbf{r}$.

A.2 - Méthode de la plus grande pente

On se donne un point de départ \mathbf{x}_0 . La première idée pour atteindre le minimum de φ à partir de ce point consiste à chercher la direction de plus grande pente (donnée par le gradient), et à descendre suivant cette direction jusqu'à atteindre un minimum (recherche linéaire). Puis on recommence le processus à partir de ce nouveau point. Cette méthode s'appelle le 'gradient simple', ou 'méthode de la plus grande pente'. Elle conduit bien au minimum de φ , mais la convergence peut être très lente si le problème est mal conditionné (cas de la vallée étroite, fig A1).

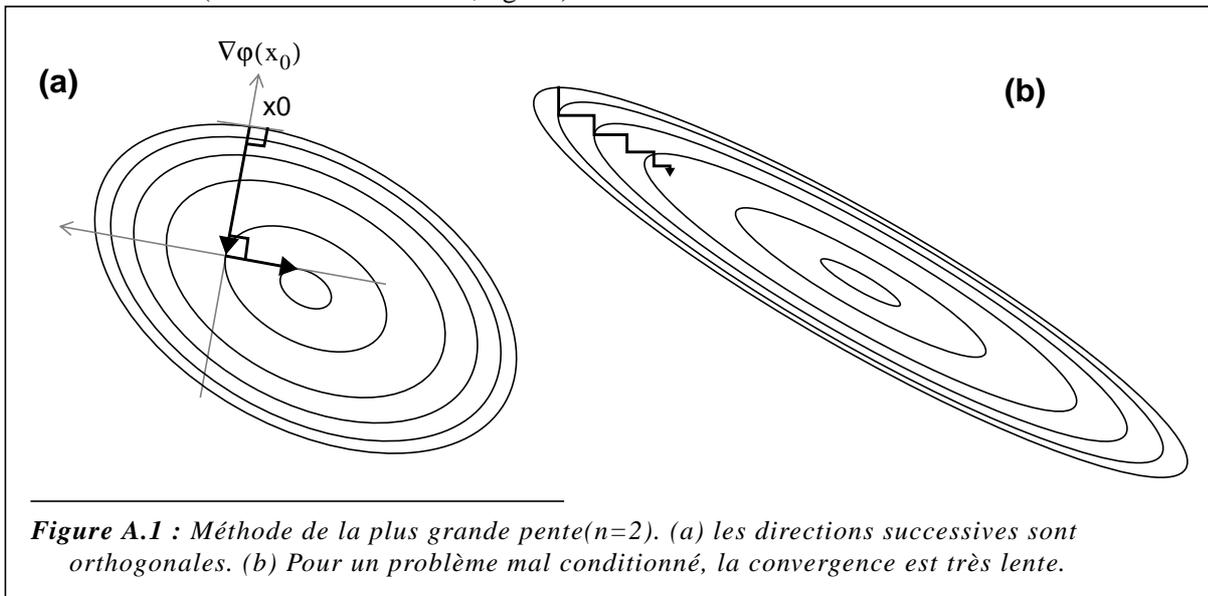


Figure A.1 : Méthode de la plus grande pente ($n=2$). (a) les directions successives sont orthogonales. (b) Pour un problème mal conditionné, la convergence est très lente.

La direction suivie à partir d'un point est le gradient, qui est orthogonal à l'hypersurface de niveau en ce point. Or la direction précédente est tangente à l'hypersurface de niveau (le point choisi constitue en effet un minimum de φ le long de cette direction): les deux directions sont donc orthogonales entre elles.

On se donne une suite de directions $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k, \dots)$, une suite de réels $(\alpha_0, \alpha_1, \dots, \alpha_k, \dots)$, et on définit la suite $(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_k, \dots)$ par:

$$\begin{cases} \mathbf{x}_0 & \text{donné} \\ \mathbf{x}_{k+1} & = \mathbf{x}_k + \alpha_k \mathbf{d}_k \end{cases}$$

On pose $\mathbf{r}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k$. On a donc: $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{d}_k$

$$\left(\begin{array}{l} \mathbf{d}_k = -\nabla\varphi(\mathbf{x}_k) = \mathbf{r}_k \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \end{array} \right) \quad \alpha_k = \frac{\mathbf{r}_k \cdot \mathbf{d}_k}{\mathbf{d}_k \cdot \mathbf{A}\mathbf{d}_k} \quad (\text{EA.4})$$

α_k est obtenu par recherche linéaire pour trouver le minimum de φ le long de la direction \mathbf{d}_k .

Lemme:

Soit \mathbf{x} un point quelconque, et \mathbf{d} une direction de recherche. Le minimum de φ le long de la droite $\mathbf{x} + \alpha\mathbf{d}$ $\alpha \in \mathbb{R}$, est obtenu par:

$$\alpha = \frac{\mathbf{r} \cdot \mathbf{d}}{\mathbf{d} \cdot \mathbf{A}\mathbf{d}} \quad (\text{EA.5})$$

Démonstration:

$$S(\alpha) = \varphi(\mathbf{x} + \alpha\mathbf{d}) = \varphi(\mathbf{0}) - \mathbf{b} \cdot (\mathbf{x} + \alpha\mathbf{d}) + \frac{1}{2}(\mathbf{x} + \alpha\mathbf{d}) \cdot \mathbf{A}(\mathbf{x} + \alpha\mathbf{d})$$

$$S(\alpha) = \varphi(\mathbf{x}) - \alpha\mathbf{b} \cdot \mathbf{d} + \alpha\mathbf{x} \cdot \mathbf{A}\mathbf{d} + \frac{1}{2}\alpha^2 \mathbf{d} \cdot \mathbf{A}\mathbf{d}$$

$$\frac{\partial}{\partial\alpha} S(\alpha) = (\mathbf{A}\mathbf{x} - \mathbf{b}) \cdot \mathbf{d} + \alpha\mathbf{d} \cdot \mathbf{A}\mathbf{d} = \alpha\mathbf{d} \cdot \mathbf{A}\mathbf{d} - \mathbf{r} \cdot \mathbf{d}$$

Le minimum est atteint pour $\frac{\partial S}{\partial\alpha} = 0$, ce qui entraîne le résultat.

Lemme:

Deux directions consécutives sont orthogonales entre elles:

$$\forall k \geq 0 \quad \mathbf{d}_{k+1} \perp \mathbf{d}_k \quad (\text{EA.6})$$

Démonstration:

$\mathbf{d}_{k+1} \cdot \mathbf{d}_k = \mathbf{r}_{k+1} \cdot \mathbf{d}_k = (\mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{d}_k) \cdot \mathbf{d}_k = \mathbf{r}_k \cdot \mathbf{d}_k - \alpha_k (\mathbf{A}\mathbf{d}_k) \cdot \mathbf{d}_k$: cette dernière quantité est nulle d'après la valeur attribuée à α_k (EA.4), ce qui prouve l'orthogonalité.

A.3 - Méthode du Gradient Conjugué

A.3.1 - Principe

Le Gradient Conjugué est une amélioration de la méthode de la plus grande pente: on cherche à construire une suite de directions telle que la recherche successive des minimums le long de chacune des

directions aboutisse au même résultat que si l'on avait cherché le minimum dans le sous-espace vectoriel engendré par ces directions:

$$\begin{aligned} \varphi(\mathbf{x}_k) &= \min_{\mathbf{x} \in (\mathbf{x}_0 + \{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{k-1}\})} \varphi(\mathbf{x}) \\ \mathbf{x}_k &= \mathbf{x}_{k-1} + \alpha_{k-1} \mathbf{d}_{k-1} = \mathbf{x}_0 + \alpha_0 \mathbf{d}_0 + \alpha_1 \mathbf{d}_1 + \dots + \alpha_{k-1} \mathbf{d}_{k-1} \end{aligned} \quad (\text{EA.7})$$

Pour vérifier cette condition, il faut que: $\forall i < k \quad \frac{\partial}{\partial \alpha_i} \varphi(\mathbf{x}_k) = 0$:

$$\begin{aligned} \frac{\partial}{\partial \alpha_i} \varphi(\mathbf{x}_k) &= \frac{\partial}{\partial \alpha_i} \left(\varphi(\mathbf{0}) - \left(\mathbf{b} \cdot \left(\mathbf{x}_0 + \sum_{j=0}^{k-1} \alpha_j \mathbf{d}_j \right) \right) + \frac{1}{2} \left(\mathbf{x}_0 + \sum_{j=0}^{k-1} \alpha_j \mathbf{d}_j \right) \cdot \mathbb{A} \left(\mathbf{x}_0 + \sum_{j=0}^{k-1} \alpha_j \mathbf{d}_j \right) \right) \\ &= -\mathbf{b}^t \mathbf{d}_i + \left(\mathbf{x}_0 + \sum_{j=0}^{k-1} \alpha_j \mathbf{d}_j \right) \cdot \mathbb{A} \mathbf{d}_i \\ &= (\mathbb{A} \mathbf{x}_0 - \mathbf{b}) \cdot \mathbf{d}_i + \sum_{j=0}^{k-1} \alpha_j (\mathbf{d}_j \cdot \mathbb{A} \mathbf{d}_i) \end{aligned}$$

La suite étant construite séquentiellement, α_i ne doit pas dépendre de $\alpha_{i+1}, \alpha_{i+2}, \dots$. Cela entraîne que $(\forall j > i \quad \mathbf{d}_j \cdot \mathbb{A} \mathbf{d}_i = 0)$: **La suite** $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k)$ **est** $\mathbb{A} \perp$. Cette condition est **nécessaire et suffisante**. On a alors: $\alpha_i = \frac{\mathbf{r}_0 \cdot \mathbf{d}_i}{\mathbf{d}_i \cdot \mathbb{A} \mathbf{d}_i}$

Lemme:

Le gradient en un point est orthogonal à toutes les directions précédentes

$$\nabla \varphi(\mathbf{x}_k) = \mathbf{r}_k \perp \{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{k-1}\} \quad (\text{EA.8})$$

Démonstration:

Soit $i < k$:

$$\begin{aligned} \mathbf{r}_k \cdot \mathbf{d}_i &= \left(\mathbf{r}_0 - \sum_{j=0}^{k-1} \alpha_j \mathbb{A} \mathbf{d}_j \right) \cdot \mathbf{d}_i \\ &= \mathbf{r}_0 \cdot \mathbf{d}_i - \alpha_i (\mathbf{d}_i \cdot \mathbb{A} \mathbf{d}_i) = \mathbf{r}_0 \cdot \mathbf{d}_i - \frac{\mathbf{r}_0 \cdot \mathbf{d}_i}{\mathbf{d}_i \cdot \mathbb{A} \mathbf{d}_i} (\mathbf{d}_i \cdot \mathbb{A} \mathbf{d}_i) = 0 \end{aligned}$$

On remarque que dans la méthode de la plus grande pente, le gradient n'est orthogonal qu'à la dernière direction.

A.3.2 - Construction de la suite d

La suite de directions est construite par les relations suivantes:

$$\left(\begin{array}{l} \mathbf{x}_0 \text{ donné} \\ \mathbf{d}_0 = \mathbf{r}_0 \\ \mathbf{d}_k = \mathbf{r}_k + \beta_k \mathbf{d}_{k-1} \quad \beta_k \text{ est choisi de manière à ce que } \mathbf{d}_k \mathbf{A} \perp \mathbf{d}_{k-1} \text{ (voir ci-dessous)} \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad \alpha_k \text{ obtenu par recherche linéaire (équ. (EA.5))} \end{array} \right) \quad (\text{EA.9})$$

Chaque nouvelle direction est une combinaison linéaire du gradient avec la direction précédente. β_k est appelé le **facteur de conjugaison**. On le choisit afin que deux directions consécutives soient **A**-orthogonales:

$$\mathbf{d}_k \cdot \mathbf{A} \mathbf{d}_{k-1} = \mathbf{0} = (\mathbf{r}_k + \beta_k \mathbf{d}_{k-1}) \cdot \mathbf{A} \mathbf{d}_{k-1}$$

$$\beta_k = - \left(\frac{\mathbf{r}_k \cdot \mathbf{A} \mathbf{d}_{k-1}}{\mathbf{d}_{k-1} \cdot \mathbf{A} \mathbf{d}_{k-1}} \right)$$

Cette formule peut s'écrire de différentes manières, qui sont toutes strictement équivalentes pour les fonctions quadratiques, mais qui produisent des résultats différents dans d'autres cas:

Formule de Fletcher and Reeves (1964) (la plus utilisée):

$$\beta_k = \frac{\mathbf{r}_k \cdot \mathbf{r}_k}{\mathbf{r}_{k-1} \cdot \mathbf{r}_{k-1}} \quad (\text{EA.10})$$

Formule de Polak and Ribiere (1969) (elle est censée donner de meilleurs résultats dans les cas non quadratiques):

$$\beta_k = \frac{\mathbf{r}_k \cdot (\mathbf{r}_k - \mathbf{r}_{k-1})}{\mathbf{r}_{k-1} \cdot \mathbf{r}_{k-1}} \quad (\text{EA.11})$$

Formule de Hestenes and Stiefel (1952):

$$\beta_k = \frac{\mathbf{r}_k \cdot (\mathbf{r}_k - \mathbf{r}_{k-1})}{\mathbf{d}_{k-1} \cdot (\mathbf{r}_k - \mathbf{r}_{k-1})} \quad (\text{EA.12})$$

Lemme:

La suite $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k)$ construite par (EA.9) est **A**-orthogonale.

Démonstration (par récurrence):

Supposons que $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{k-1})$ soit **A**-orthogonale, et calculons $\mathbf{d}_k \cdot \mathbf{A} \mathbf{d}_i$ pour $i < k$

$$i = k-1 \quad \mathbf{d}_k \cdot \mathbf{A} \mathbf{d}_i = \mathbf{0} \text{ par construction}$$

$$i < k-1 \quad \mathbf{d}_k \cdot \mathbf{A} \mathbf{d}_i = \mathbf{r}_k \cdot \mathbf{A} \mathbf{d}_i + \beta_k \mathbf{d}_{k-1} \cdot \mathbf{A} \mathbf{d}_i$$

Montrons que $\mathbf{A} \mathbf{d}_i \in \{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{i+1}\}$, ce qui, d'après l'orthogonalité entre le gradient et les directions de rangs inférieurs, entraîne le résultat désiré, à savoir que

($i < k - 1$ $\mathbf{d}_k \cdot \mathbf{A}\mathbf{d}_i = \mathbf{0}$), donc que $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k)$ est \mathbf{A} -orthogonale, complétant ainsi la récurrence.

On procède à nouveau par récurrence:

$$\mathbf{d}_1 = \mathbf{r}_1 + \beta_1 \mathbf{d}_0 = (\mathbf{r}_0 - \alpha_0 \mathbf{A}\mathbf{d}_0) + \beta_1 \mathbf{d}_0$$

$$\alpha_0 \mathbf{A}\mathbf{d}_0 = (1 + \beta_1) \mathbf{d}_0 - \mathbf{d}_1$$

Ce qui signifie, si l'on exclu le cas $\alpha_0=0$ (minimum atteint), que $\mathbf{A}\mathbf{d}_0 \in \{\mathbf{d}_0, \mathbf{d}_1\}$

supposons que $\mathbf{A}\mathbf{d}_{i-1} \in \{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_i\}$:

$$\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \beta_{i+1} \mathbf{d}_i = \left(\mathbf{r}_0 - \sum_{j=0}^i \alpha_j \mathbf{A}\mathbf{d}_j \right) + \beta_{i+1} \mathbf{d}_i$$

$$\alpha_i \mathbf{A}\mathbf{d}_i = \mathbf{d}_0 - \sum_{j=0}^{i-1} \alpha_j \mathbf{A}\mathbf{d}_j + \beta_{i+1} \mathbf{d}_i - \mathbf{d}_{i+1}$$

D'après l'hypothèse de récurrence, le second terme de la dernière équation appartient à $\{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_i\}$. Donc, en excluant le cas $\alpha_i=0$ (minimum atteint), on a bien

$\mathbf{A}\mathbf{d}_i \in \{\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_{i+1}\}$, ce qui complète ainsi la récurrence.

Etant \mathbf{A} -orthogonale (condition suffisante), la suite $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k)$ répond au critère défini en (EA.7).

Lemme:

La suite des gradient $(\mathbf{r}_0, \mathbf{r}_1, \dots, \mathbf{r}_k, \dots)$ est orthogonale.

Démonstration:

$$i < k \quad \mathbf{r}_k \cdot \mathbf{r}_i = \mathbf{r}_k \cdot (\mathbf{d}_i - \beta_i \mathbf{d}_{i-1}) = \mathbf{0} \quad (\text{d'après (EA.8)})$$

Le côté 'magique' du gradient conjugué tient à la propriété suivante: chaque itération consiste à chercher le minimum non pas seulement dans la direction du gradient, mais dans le plan défini par le gradient et la direction de descente précédente. Il s'agit donc d'une recherche bilinéaire, plus ou moins occultée par la formule de conjugaison qui ramène à une simple recherche linéaire.

On pourrait alors penser qu'en utilisant les directions calculées aux itérations précédentes (avec des recherches tri- ou quadri-linéaires,...), on améliorerait encore l'optimisation. Or ce n'est pas le cas, car la dernière direction calculée contient déjà des informations relatives à toutes les directions précédentes.

A.3.3 - Préconditionnement

Le preconditionnement consiste à choisir comme direction de base non pas le gradient, mais une direction calculée d'après le gradient et qui 'pointe mieux' vers le minimum à atteindre. Le preconditionnement peut être abordé avec des points de vues différents, mais qui au final aboutissent tous au même résultat: changement de variable, changement de produit scalaire, ou simplement application d'un opérateur arbitraire sur le gradient.

Nous l'abordons ici du point de vue des changements de variables. Au lieu de résoudre le système $\mathbf{A}\mathbf{x} = \mathbf{b}$, on veut résoudre le système strictement équivalent: $\mathbf{A}\mathbf{W}\hat{\mathbf{x}} = \mathbf{b}$, en posant le changement de variable $\mathbf{x} = \mathbf{W}\hat{\mathbf{x}}$, où \mathbf{W} est une matrice définie positive. On multiplie encore le système à gauche par

\mathbf{W}^t , et on pose $\hat{\mathbf{A}} = \mathbf{W}^t \mathbf{A} \mathbf{W}$ et $\hat{\mathbf{b}} = \mathbf{W}^t \mathbf{b}$. $\hat{\mathbf{A}}$ est définie, positive et symétrique, ce qui permet d'appliquer directement les formules du gradient conjugué (EA.9) au nouveau système:

$$\hat{\mathbf{A}} \hat{\mathbf{x}} = \hat{\mathbf{b}} \quad (\text{EA.13})$$

On construit donc une suite de directions $(\hat{\mathbf{d}}_0, \hat{\mathbf{d}}_1, \dots, \hat{\mathbf{d}}_k)$:

$$\left\{ \begin{array}{l} \hat{\mathbf{x}}_0 = \mathbf{W}^{-1} \mathbf{x}_0 \\ \hat{\mathbf{d}}_0 = \hat{\mathbf{r}}_0 \\ \hat{\mathbf{d}}_k = \hat{\mathbf{r}}_k + \hat{\beta}_k \hat{\mathbf{d}}_{k-1} \quad \hat{\beta}_k = \frac{\hat{\mathbf{r}}_k \cdot (\hat{\mathbf{r}}_k - \hat{\mathbf{r}}_{k-1})}{\hat{\mathbf{r}}_{k-1} \cdot \hat{\mathbf{r}}_{k-1}} \\ \hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_k + \hat{\alpha}_k \hat{\mathbf{d}}_k \quad \hat{\alpha}_k = \frac{\hat{\mathbf{r}}_k \cdot \hat{\mathbf{d}}_k}{\hat{\mathbf{d}}_k \cdot \hat{\mathbf{A}} \hat{\mathbf{d}}_k} \end{array} \right.$$

Les directions exprimées par rapport à la variable d'origine \mathbf{x} s'obtiennent par: $\mathbf{W} \hat{\mathbf{d}} = \mathbf{d}$. Le gradient quant à lui s'écrit simplement: $\hat{\mathbf{r}} = \hat{\mathbf{b}} - \hat{\mathbf{A}} \hat{\mathbf{x}} = \mathbf{W}^t (\mathbf{b} - \mathbf{A} \mathbf{W} \mathbf{W}^{-1} \mathbf{x}) = \mathbf{W}^t \mathbf{r}$. On pose par ailleurs $\mathbf{Q} = \mathbf{W} \mathbf{W}^t$ (\mathbf{Q} est définie, positive, symétrique). Ce qui donne, une fois que l'on repasse à la variable d'origine:

$$\left\{ \begin{array}{l} \mathbf{x}_0 \text{ donné} \\ \mathbf{d}_0 = \mathbf{Q} \mathbf{r}_0 \\ \mathbf{d}_k = \mathbf{Q} \mathbf{r}_k + \beta_k \mathbf{d}_{k-1} \quad \beta_k = \frac{\mathbf{r}_k \cdot \mathbf{Q} (\mathbf{r}_k - \mathbf{r}_{k-1})}{\mathbf{r}_{k-1} \cdot \mathbf{Q} \mathbf{r}_{k-1}} \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad \alpha_k = \frac{\mathbf{r}_k \cdot \mathbf{d}_k}{\mathbf{d}_k \cdot \mathbf{A} \mathbf{d}_k} \end{array} \right. \quad (\text{EA.14})$$

On peut montrer également qu'en choisissant arbitrairement la matrice \mathbf{Q} définie, positive, symétrique, la suite de direction $(\mathbf{d}_0, \mathbf{d}_1, \dots, \mathbf{d}_k)$ ainsi construite est \mathbf{A} -orthogonale, satisfaisant ainsi les critères du gradient conjugué.

Le hessien associé au nouveau problème est $\hat{\mathbf{A}} = \mathbf{W}^t \mathbf{A} \mathbf{W}$. On sait qu'un problème de ce type est d'autant mieux conditionné que son hessien est proche de l'identité. Cela suggère un choix pour le changement de variable: il faut que $\mathbf{W} \mathbf{W}^t = \mathbf{Q} \approx \mathbf{A}^{-1}$, c'est à dire qu'il faut arriver à trouver une bonne approximation de l'inverse du hessien initial (\mathbf{A}).

A.3.4 - Produit scalaire quelconque

On veut voir maintenant comment s'écrivent les formules du gradient conjugué lorsque l'on utilise un produit scalaire plus général que le produit scalaire euclidien: Soit \mathbf{B} une matrice définie, positive, symétrique, et le produit scalaire associé:

$$(\mathbf{x}, \mathbf{y}) = \mathbf{x} \cdot \mathbf{B} \mathbf{y} \quad (\text{EA.15})$$

On cherche à minimiser:

$$\varphi(\mathbf{x}) = \varphi(\mathbf{0}) - (\mathbf{b}, \mathbf{x}) + \frac{1}{2}(\mathbf{x}, \mathbf{A}\mathbf{x}) = \varphi(\mathbf{0}) - (\mathbf{B}\mathbf{b}) \cdot \mathbf{x} + \frac{1}{2}\mathbf{x} \cdot (\mathbf{B}\mathbf{A})\mathbf{x}$$

Ce qui revient à résoudre le système $\mathbf{B}\mathbf{A}\mathbf{x} = \mathbf{B}\mathbf{b}$. Ce système est strictement équivalent au système précédent $\mathbf{A}\mathbf{x} = \mathbf{b}$. On pose donc à nouveau $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{x}$. Le gradient et le hessien du nouveau problème sont:

$$\nabla\varphi(\mathbf{x}) = (\mathbf{B}\mathbf{A})\mathbf{x} - \mathbf{B}\mathbf{b} = -\mathbf{B}\mathbf{r}$$

$$\nabla^2\varphi(\mathbf{x}) = \mathbf{B}\mathbf{A}$$

L'application des formules de gradient conjugué dans ce cas donne:

$$\left\{ \begin{array}{l} \mathbf{x}_0 \text{ donné} \\ \mathbf{d}_0 = \mathbf{B}\mathbf{r}_0 \\ \mathbf{d}_k = \mathbf{B}\mathbf{r}_k + \beta_k \mathbf{d}_{k-1} \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \end{array} \right. \quad \left\{ \begin{array}{l} \beta_k = \frac{(\mathbf{B}\mathbf{r}_k) \cdot (\mathbf{B}\mathbf{r}_k - \mathbf{B}\mathbf{r}_{k-1})}{(\mathbf{B}\mathbf{r}_{k-1}) \cdot \mathbf{B}\mathbf{r}_{k-1}} = \frac{(\mathbf{r}_k, \mathbf{B}(\mathbf{r}_k - \mathbf{r}_{k-1}))}{(\mathbf{r}_{k-1}, \mathbf{B}\mathbf{r}_k)} \\ \alpha_k = \frac{(\mathbf{B}\mathbf{r}_k) \cdot \mathbf{d}_k}{\mathbf{d}_k \cdot (\mathbf{B}\mathbf{A})\mathbf{d}_k} = \frac{(\mathbf{r}_k, \mathbf{d}_k)}{(\mathbf{d}_k, \mathbf{A}\mathbf{d}_k)} \end{array} \right. \quad (\text{EA.16})$$

Le gradient correspond ici à la direction de plus grande pente pour le produit scalaire euclidien. Le fait d'employer un autre produit scalaire change cette direction de plus grande pente. En effet:

$$\begin{aligned} \varphi(\mathbf{x} + \delta\mathbf{x}) &= \varphi(\mathbf{x}) + (-\mathbf{B}\mathbf{r}) \cdot \delta\mathbf{x} + (\text{terme de second ordre...}) \\ &= \varphi(\mathbf{x}) + (-\mathbf{r}, \delta\mathbf{x}) + (\text{terme de second ordre...}) \end{aligned}$$

La direction de plus grande pente associée par rapport au nouveau produit scalaire est donc non plus $(-\mathbf{B}\mathbf{r})$, mais $(-\mathbf{r})$. Cela nous incite à préconditionner le gradient conjugué afin d'en tenir compte. En préconditionnant avec $\mathbf{Q} = \mathbf{B}^{-1}$, on obtient:

$$\left\{ \begin{array}{l} \mathbf{x}_0 \text{ donné} \\ \mathbf{d}_0 = \mathbf{r}_0 \\ \mathbf{d}_k = \mathbf{r}_k + \beta_k \mathbf{d}_{k-1} \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \end{array} \right. \quad \left\{ \begin{array}{l} \beta_k = \frac{(\mathbf{r}_k, \mathbf{r}_k - \mathbf{r}_{k-1})}{(\mathbf{r}_{k-1}, \mathbf{r}_k)} \\ \alpha_k = \frac{(\mathbf{r}_k, \mathbf{d}_k)}{(\mathbf{d}_k, \mathbf{A}\mathbf{d}_k)} \end{array} \right. \quad (\text{EA.17})$$

Les équations sont en fait les mêmes que précédemment, en remplaçant formellement le produit scalaire euclidien par le nouveau produit scalaire.

A.3.5 - Recherches bi/tri.. -linéaires

La formule de conjugaison suivie d'une recherche linéaire cache en fait une recherche bi-linéaire sur le gradient et la direction de descente précédente. Cette simplification provient de la nature quadratique du problème et de la connaissance du gradient exact en tous points. Lorsque l'on s'écarte de ces hypothèses, on a le choix entre utiliser des formules de conjugaisons alternatives ((EA.11) ou (EA.12)), ou effectuer une vraie recherche bi-linéaire. C'est le cas lorsque l'on utilise une autre direction que le

gradient vrai (soit que le calcul de cette direction soit plus rapide que celui du gradient, soit que l'on pense que cette direction mène plus rapidement au minimum que le gradient vrai), ou si le problème à résoudre n'est pas quadratique mais approximable à chaque itération par un problème quadratique (auquel cas les paramètres \underline{A} et \mathbf{b} changent à chaque itération).

Si en point \mathbf{x} on se donne deux direction $\mathbf{g1}$ et $\mathbf{g2}$ quelconques. On veut trouver le minimum dans le plan $\mathbf{x} + \{\mathbf{g1}, \mathbf{g2}\}$:

$$\varphi(\mathbf{x} + \alpha_1 \mathbf{g1} + \alpha_2 \mathbf{g2}) =$$

$$\varphi(\mathbf{x}) + \alpha_1 \mathbf{x} \cdot \underline{A} \mathbf{g1} + \alpha_2 \mathbf{x} \cdot \underline{A} \mathbf{g2} + \frac{\alpha_1^2}{2} \mathbf{g1} \cdot \underline{A} \mathbf{g1} + \alpha_1 \alpha_2 \mathbf{g1} \cdot \underline{A} \mathbf{g2} + \frac{\alpha_2^2}{2} \mathbf{g2} \cdot \underline{A} \mathbf{g2} - \alpha_1 \mathbf{b} \cdot \mathbf{g1} - \alpha_2 \mathbf{b} \cdot \mathbf{g2}$$

On calcule les dérivées de cette quantité par rapport aux deux scalaires α_1 et α_2 :

$$\frac{\partial}{\partial \alpha_1} \varphi(\mathbf{x} + \alpha_1 \mathbf{g1} + \alpha_2 \mathbf{g2}) = \mathbf{x} \cdot \underline{A} \mathbf{g1} + \alpha_1 \mathbf{g1} \cdot \underline{A} \mathbf{g1} + \alpha_2 \mathbf{g1} \cdot \underline{A} \mathbf{g2} - \mathbf{b} \cdot \mathbf{g1}$$

$$\frac{\partial}{\partial \alpha_2} \varphi(\mathbf{x} + \alpha_1 \mathbf{g1} + \alpha_2 \mathbf{g2}) = \mathbf{x} \cdot \underline{A} \mathbf{g2} + \alpha_2 \mathbf{g2} \cdot \underline{A} \mathbf{g2} + \alpha_1 \mathbf{g1} \cdot \underline{A} \mathbf{g2} - \mathbf{b} \cdot \mathbf{g2}$$

Ces dérivées doivent être nulles au minimum cherché, ce qui conduit à résoudre un petit système linéaire 2×2 pour trouver α_1 et α_2 :

$$\begin{bmatrix} \mathbf{g1} \cdot \underline{A} \mathbf{g1} & \mathbf{g1} \cdot \underline{A} \mathbf{g2} \\ \mathbf{g1} \cdot \underline{A} \mathbf{g2} & \mathbf{g2} \cdot \underline{A} \mathbf{g2} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b} \cdot \mathbf{g1} - \mathbf{x} \cdot \underline{A} \mathbf{g1} \\ \mathbf{b} \cdot \mathbf{g2} - \mathbf{x} \cdot \underline{A} \mathbf{g2} \end{bmatrix} \quad (\text{EA.18})$$

On peut ajouter des directions de recherche supplémentaires. Il faut alors résoudre un système linéaire de la taille du nombre de directions. Si on se donne N directions $\mathbf{g1}, \mathbf{g2}, \dots, \mathbf{gN}$, il faut résoudre:

$$\begin{bmatrix} \mathbf{g1} \cdot \underline{A} \mathbf{g1} & \mathbf{g1} \cdot \underline{A} \mathbf{g2} & \dots & \dots & \mathbf{g1} \cdot \underline{A} \mathbf{gN} \\ \mathbf{g1} \cdot \underline{A} \mathbf{g2} & \mathbf{g2} \cdot \underline{A} \mathbf{g2} & & & \dots \\ \dots & & \cdot & & \dots \\ \dots & & & \cdot & \dots \\ \mathbf{g1} \cdot \underline{A} \mathbf{gN} & \dots & \dots & \dots & \mathbf{gN} \cdot \underline{A} \mathbf{gN} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \dots \\ \dots \\ \alpha_N \end{bmatrix} = \begin{bmatrix} \mathbf{b} \cdot \mathbf{g1} - \mathbf{x} \cdot \underline{A} \mathbf{g1} \\ \mathbf{b} \cdot \mathbf{g2} - \mathbf{x} \cdot \underline{A} \mathbf{g2} \\ \dots \\ \dots \\ \mathbf{b} \cdot \mathbf{gN} - \mathbf{x} \cdot \underline{A} \mathbf{gN} \end{bmatrix} \quad (\text{EA.19})$$

Annexe B:

Convergence des algorithmes IRLS

On reprend ici la démonstration de convergence de Byrd and Pyne (1979, non disponible), reprise et généralisée dans Bube and Langan (1997).

Soit un système linéaire sur-déterminé à résoudre:

$$\mathbf{Ax} = \mathbf{b} \tag{EA.20}$$

Le résidu du système est défini par $\mathbf{r} = \mathbf{b} - \mathbf{Ax}$. On veut minimiser la fonction coût:

$$\varphi(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{r}_i) \tag{EA.21}$$

Où les $f_i()$ sont des fonctions scalaires.

On pose les fonctions $\begin{cases} g_i(r \neq 0) = \frac{f_i'(r)}{r} \\ g_i(0) = f_i''(0) \end{cases}$ et on fait les hypothèses suivantes:

- (i) f_i de classe C1
- (ii) $f_i''(0)$ existe et est finie
- (iii) f_i strictement convexe
- (iv) f_i paire
- (v) $g_i(r)$ décroissante pour $(r \geq 0)$

(EA.22)

Des hypothèses précédentes, on déduit facilement que: $f(r \neq 0) > f(0)$ et $\lim_{r \rightarrow \infty} f(r) = +\infty$

La fonction coût a donc un minimum unique, au point \mathbf{x}_* .

Soit la matrice diagonale dépendant de \mathbf{x} définie par:

$$(\mathbf{W}(\mathbf{x}))_{ii} = g_i(\mathbf{r}_i) \tag{EA.23}$$

Le gradient de la fonction coût est:

$$(\nabla\varphi(\mathbf{x}))_j = \sum_{i=1}^n \frac{\partial}{\partial \mathbf{x}_j} f_i(\mathbf{r}_i) = \sum_{i=1}^n f_i'(\mathbf{r}_i) \frac{\partial \mathbf{r}_i}{\partial \mathbf{x}_j} = \sum_{i=1}^n f_i'(\mathbf{r}_i) \frac{\partial \mathbf{r}_i}{\partial \mathbf{x}_j}$$

Or le résidu est égal à: $\mathbf{r}_i = (\mathbf{b} - \mathbf{A}\mathbf{x})_i = b_i - \sum_{l=1}^n \mathbf{A}_{il}\mathbf{x}_l$. Ce qui donne pour le gradient:

$$(\nabla\varphi(\mathbf{x}))_j = - \sum_{i=1}^n f_i'(\mathbf{r}_i) \mathbf{A}_{ij} = - \sum_{i=1}^n \mathbf{r}_i g_i(\mathbf{r}_i) \mathbf{A}_{ij} = - \sum_{i=1}^n \mathbf{A}_{ji}^T ((\mathbf{W}(\mathbf{x}))_{ii} \mathbf{r}_i)$$

D'où:

$$\nabla\varphi(\mathbf{x}) = -\mathbf{A}^T \mathbf{W}(\mathbf{x}) \mathbf{r} = \mathbf{A}^T \mathbf{W}(\mathbf{x}) (\mathbf{A}\mathbf{x} - \mathbf{b}) \tag{EA.24}$$

le minimum est atteint lorsque le gradient s'annule. \mathbf{x}_* est donc solution du système:

$$\mathbf{A}^T \mathbf{W}(\mathbf{x}) \mathbf{A} \mathbf{x} = \mathbf{A}^T \mathbf{W}(\mathbf{x}) \mathbf{b} \tag{EA.25}$$

Ce système est non-linéaire à cause de la matrice \mathbf{W} qui dépend de \mathbf{x} .

L'algorithme IRLS consiste à construire à partir d'une valeur initiale arbitraire \mathbf{x}^0 une suite $\{\mathbf{x}^k\}_{k=1, \dots}$ dont chaque élément \mathbf{x}^k est solution du système linéaire:

$$\begin{aligned} \mathbf{A}^T \mathbf{W}^k \mathbf{A} \mathbf{x} &= \mathbf{A}^T \mathbf{W}^k \mathbf{b} \\ \mathbf{x}^k &= (\mathbf{A}^T \mathbf{W}^{(k)} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W}^{(k)} \mathbf{b} \end{aligned} \tag{EA.26}$$

avec $\mathbf{W}^{(k)} = \mathbf{W}(\mathbf{x}^{k-1})$.

On veut montrer que cette suite converge vers \mathbf{x}_* .

La résolution de (EA.26) correspond à la minimisation de la fonction coût:

$$\psi^k(\mathbf{x}) = \frac{1}{2} \mathbf{r} \cdot \mathbf{W}^{(k)} \mathbf{r} - \frac{1}{2} \mathbf{r}^{k-1} \cdot \mathbf{W}^{(k)} \mathbf{r}^{k-1} + \varphi(\mathbf{x}^{k-1}) \quad (\text{EA.27})$$

dont le gradient est:

$$\nabla \psi^k(\mathbf{x}) = -\mathbf{A}^T \mathbf{W}^{(k)} \mathbf{r} \quad (\text{EA.28})$$

Lemme 1:

ψ^k est tangente à φ en $\mathbf{x} = \mathbf{x}^{k-1}$, et la domine:

Démonstration:

La tangence en $\mathbf{x} = \mathbf{x}^{k-1}$ est triviale (équ. (EA.24), (EA.27), (EA.28)):

$$\boxed{\psi^k(\mathbf{x}^{k-1}) = \varphi(\mathbf{x}^{k-1}) \quad \nabla \psi^k(\mathbf{x}^{k-1}) = \nabla \varphi(\mathbf{x}^{k-1})} \quad (\text{EA.29})$$

Il reste à montrer que ψ^k domine φ :

$$\begin{aligned} \psi^k(\mathbf{x}) - \varphi(\mathbf{x}) &= \int_{\mathbf{x}^{k-1}}^{\mathbf{x}} (\nabla \psi^k(\mathbf{y}) - \nabla \varphi(\mathbf{y})) \cdot d\mathbf{y} \\ &= \int_{\mathbf{x}^{k-1}}^{\mathbf{x}} (-\mathbf{A}^T \mathbf{W}^{(k)} \mathbf{s} + \mathbf{A}^T \mathbf{W}(\mathbf{y}) \mathbf{s}) \cdot d\mathbf{y} \\ &= \int_{\mathbf{x}^{k-1}}^{\mathbf{x}} (\mathbf{W}^{(k)} \mathbf{s} - \mathbf{W}(\mathbf{y}) \mathbf{s}) \cdot (-\mathbf{A} d\mathbf{y}) \end{aligned}$$

où $\mathbf{s} = \mathbf{b} - \mathbf{A} \mathbf{y}$. Si on opère le changement de variable $\mathbf{r} = \mathbf{b} - \mathbf{A} \mathbf{x}$, on a $d\mathbf{r} = -\mathbf{A} d\mathbf{x}$:

$$\psi^k(\mathbf{x}) - \varphi(\mathbf{x}) = \int_{\mathbf{r}^{k-1}}^{\mathbf{r}} (\mathbf{W}^{(k)} \mathbf{s} - \mathbf{W}(\mathbf{s}) \mathbf{s}) \cdot d\mathbf{s}$$

$$(\mathbf{W}(\mathbf{s}))_{ii} = g_i(s_i) \quad , \text{ d'où: } (\mathbf{W}^{(k)} \mathbf{s} - \mathbf{W}(\mathbf{s}) \mathbf{s})_i = s_i (g_i(s_i^{k-1}) - g_i(s_i)) \quad .$$

Les fonction g_i étant paires et décroissantes pour $r > 0$, on peut en déduire que

$$\int_{\mathbf{r}_i^{k-1}}^{\mathbf{r}_i} s_i (g_i(s_i^{k-1}) - g_i(s_i)) ds_i \geq 0 \quad . \text{ D'où le résultat recherché:}$$

$$\boxed{\psi^k(\mathbf{x}) - \varphi(\mathbf{x}) \geq 0} \quad (\text{EA.30})$$

Lemme 2:

la suite $\{\varphi(\mathbf{x}^k)\}_k$ est convergente.

Démonstration:

\mathbf{x}^k étant obtenu par minimisation de ψ^k , on peut tirer de (EA.29) et (EA.30) l'inégalité:

$$\varphi(\mathbf{x}^{k-1}) = \psi^k(\mathbf{x}^{k-1}) \geq \psi^k(\mathbf{x}^k) \geq \varphi(\mathbf{x}^k) \quad (\text{EA.31})$$

La suite $\{\varphi(\mathbf{x}^k)\}_k$ est donc décroissante, et comme la fonction φ est minorée par hypothèse, elle converge.

Lemme 3:

$$\lim_{k \rightarrow \infty} \|\nabla\varphi(\mathbf{x}^k)\|^2 = 0$$

Démonstration:

D'après (EA.26):

$$\begin{aligned} \mathbf{x}^k &= (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \underline{\mathbf{A}})^{-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{b} = \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{b} \\ &= \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} (\mathbf{r}^{k-1} + \underline{\mathbf{A}} \mathbf{x}^{k-1}) \\ &= \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1} + \mathbf{x}^{k-1} \end{aligned}$$

avec $\underline{\mathbf{B}}^{(k)} = \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \underline{\mathbf{A}}$

Donc:

$$\mathbf{r}^k - \mathbf{r}^{k-1} = -\underline{\mathbf{A}}(\mathbf{x}^k - \mathbf{x}^{k-1}) = -\underline{\mathbf{A}} \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}$$

On introduit cette dernière relation dans (EA.27):

$$\begin{aligned} \psi^k(\mathbf{x}^k) &= \frac{1}{2} \mathbf{r}^k \cdot \underline{\mathbf{W}}^{(k)} \mathbf{r}^k - \frac{1}{2} \mathbf{r}^{k-1} \cdot \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1} + \varphi(\mathbf{x}^{k-1}) \\ &= -\mathbf{r}^{k-1} \cdot \underline{\mathbf{W}}^{(k)} (\underline{\mathbf{A}} \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) + \frac{1}{2} (\underline{\mathbf{A}} \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot \underline{\mathbf{W}}^k (\underline{\mathbf{A}} \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) + \varphi(\mathbf{x}^{k-1}) \\ &= -(\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot \underline{\mathbf{B}}^{(k)-1} (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) + \frac{1}{2} (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot (\underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) + \varphi(\mathbf{x}^{k-1}) \\ &= -\frac{1}{2} (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot \underline{\mathbf{B}}^{(k)-1} (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) + \varphi(\mathbf{x}^{k-1}) \end{aligned}$$

D'après (EA.28):

$$\begin{aligned} \|\nabla\psi^k(\mathbf{x}^{k-1})\|^2 &= \|\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}\|^2 = (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \\ &= (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \\ &= (\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot (\underline{\mathbf{B}}^{(k)} \underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \\ &\leq \|\underline{\mathbf{B}}^{(k)}\| \left((\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \cdot (\underline{\mathbf{B}}^{(k)-1} \underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \mathbf{r}^{k-1}) \right) \end{aligned}$$

Ce dernier produit scalaire se retrouve dans les équations ci-dessus, d'où:

$$\|\nabla\psi^k(\mathbf{x}^{k-1})\|^2 \leq \|\underline{\mathbf{B}}^{(k)}\| 2(\varphi(\mathbf{x}^{k-1}) - \psi^k(\mathbf{x}^k))$$

Donc, d'après les relations (EA.29) et (EA.30):

$$\begin{aligned} \|\nabla\varphi(\mathbf{x}^{k-1})\|^2 &= \|\nabla\psi^k(\mathbf{x}^{k-1})\|^2 \\ &\leq 2\|\underline{\mathbf{B}}^{(k)}\| (\varphi(\mathbf{x}^{k-1}) - \psi^k(\mathbf{x}^k)) \\ &\leq 2\|\underline{\mathbf{B}}^{(k)}\| (\varphi(\mathbf{x}^{k-1}) - \varphi(\mathbf{x}^k)) \end{aligned}$$

Par hypothèse les fonctions g_i sont bornées, donc $\|\underline{\mathbf{W}}^{(k)}\|$ est bornée également. Il existe donc un réel M tel que: $\|\underline{\mathbf{B}}^{(k)}\| = \|\underline{\mathbf{A}}^T \underline{\mathbf{W}}^{(k)} \underline{\mathbf{A}}\| \leq M$. On peut donc écrire:

$$\|\nabla\varphi(\mathbf{x}^{k-1})\|^2 \leq 2M(\varphi(\mathbf{x}^{k-1}) - \varphi(\mathbf{x}^k))$$

La suite $\{\varphi(\mathbf{x}^k)\}_k$ étant convergente (lemme 2), on a en particulier

$$\lim_{k \rightarrow \infty} (\varphi(\mathbf{x}^{k-1}) - \varphi(\mathbf{x}^k)) = 0, \text{ donc } \lim_{k \rightarrow \infty} \|\nabla\varphi(\mathbf{x}^k)\|^2 = 0, \text{ c.q.f.d.}$$

La suite $\{\varphi(\mathbf{x}^k)\}_k$ est convergente, la suite $\{\nabla\varphi(\mathbf{x}^k)\}_k$ tend vers zero, $\nabla\varphi$ est une fonction continue qui a pour unique zéro \mathbf{x}_* , et $\lim_{\|\mathbf{x}\| \rightarrow \infty} \varphi(\mathbf{x}) = \infty$ par hypothèse. On en déduit que $\lim_{k \rightarrow \infty} \mathbf{x}^k = \mathbf{x}_*$.

Annexe C: Matrices ‘Bloc-Toeplitz’

C.1 - Système de Toeplitz

La résolution d’un système linéaire du type suivant est très courante dans les problèmes de traitement du signal:

$$\begin{bmatrix} r_0 & r_1 & r_2 & \cdot & \cdot & r_n \\ r_1 & r_0 & \cdot & \cdot & \cdot & r_{n-1} \\ & & & & r_{n-2} & \cdot \\ & & & & \cdot & \cdot \\ & & & & \cdot & \cdot \\ r_n & r_{n-1} & \cdot & \cdot & \cdot & r_0 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ \cdot \\ \cdot \\ \cdot \\ f_n \end{bmatrix} = \begin{bmatrix} g_0 \\ g_1 \\ \cdot \\ \cdot \\ \cdot \\ g_n \end{bmatrix} \tag{EA.32}$$

La matrice, carrée, réelle, constante le long de chaque diagonale, et symétrique, est une *matrice de Toeplitz*. Levinson a, en 1947, développé l’algorithme très connu qui porte son nom, pour résoudre d’une manière efficace ce type de système. Cet algorithme requiert en effet de l’ordre de n^2 opérations seulement, au lieu des n^3 habituelles pour une matrice quelconque.

Cette méthode est bien documentée dans la littérature (voir par exemple Robinson and Treitel, 1980, p.163). On peut sans problème l’étendre à des données complexes, où la matrice est hermitienne. Elle peut également se généraliser pour des matrices non symétriques (voir par exemple la méthode présentée dans Press et al., 1992, p.85).

C.2 - Matrice Toeplitz par blocs

On cherche ici à étendre la méthode présentée dans Robinson and Treitel en remplaçant les scalaires (r_0, r_1, \dots, r_n) par des matrices carrées complexes (toutes de même dimension). La seule complication par rapport à la méthode originale provient de la non-commutativité des produits de matrices.

On définit donc la matrice suivante:

$$\mathbf{Q}_n = \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_1^* & \cdot & \cdot & \mathbf{R}_n^* \\ \mathbf{R}_1 & \mathbf{R}_0 & & & \\ \cdot & & & & \\ \cdot & & & & \\ \mathbf{R}_n & & & & \mathbf{R}_0 \end{bmatrix} \quad (\text{EA.33})$$

\mathbf{Q}_n est composée de matrices carrées complexes $(\mathbf{R}_0, \mathbf{R}_1, \dots, \mathbf{R}_n)$, toutes de même dimension p . Sur chaque diagonale de \mathbf{Q}_n , on retrouve la même matrice \mathbf{R}_1 . \mathbf{Q}_n est une matrice que l'on pourrait donc qualifier de '**Bloc-Toeplitz**', par analogie avec une matrice de Toeplitz, puisque les scalaires sont remplacés par des matrices. On suppose en outre que \mathbf{R}_0 est hermitienne ($\mathbf{R}_0^* = \mathbf{R}_0$), si bien que \mathbf{Q}_n est elle-même hermitienne:

$$\mathbf{Q}_n^* = \mathbf{Q}_n \quad (\text{EA.34})$$

Soit \mathbf{g} un vecteur de dimension $(n.p)$, on veut résoudre le système:

$$\mathbf{Q}_n \cdot \mathbf{f} = \mathbf{g} \quad (\text{EA.35})$$

On commence par découper \mathbf{f} et \mathbf{g} en n sous-vecteurs de longueurs p :

$$\mathbf{f} = \begin{bmatrix} \mathbf{f}_0 \\ \mathbf{f}_1 \\ \cdot \\ \cdot \\ \mathbf{f}_n \end{bmatrix} \quad \mathbf{g} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \cdot \\ \cdot \\ \mathbf{g}_n \end{bmatrix} \quad (\text{EA.36})$$

Comme dans l'algorithme classique, on procède par récursivité: on suppose qu'à l'étape n , on connaît les quantités suivantes:

$$\mathbf{f}^n = \begin{bmatrix} \mathbf{f}_0^n \\ \mathbf{f}_1^n \\ \cdot \\ \cdot \\ \mathbf{f}_n^n \end{bmatrix} \quad \gamma_n \quad \mathbf{a}^n = \begin{bmatrix} \mathbf{a}_0^n \\ \mathbf{a}_1^n \\ \cdot \\ \cdot \\ \mathbf{a}_n^n \end{bmatrix} \quad \mathbf{b}^n = \begin{bmatrix} \mathbf{b}_0^n \\ \mathbf{b}_1^n \\ \cdot \\ \cdot \\ \mathbf{b}_n^n \end{bmatrix} \quad \alpha_n, \beta_n, \delta_n, \varepsilon_n$$

où \mathbf{f}_i^n, γ_n sont des vecteurs de longueur p , $\mathbf{a}_i^n, \mathbf{b}_i^n, \alpha_n, \beta_n, \delta_n, \varepsilon_n$ sont des matrices de dimension $(p.p)$.

Par définition, on suppose que ces quantités satisfont les relations¹:

$$\mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{f}_0^n \\ \mathbf{f}_1^n \\ \cdot \\ \cdot \\ \mathbf{f}_n^n \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \cdot \\ \cdot \\ \mathbf{g}_n \\ \gamma_n \end{bmatrix} \quad \mathbf{Q}_{n+1} \begin{bmatrix} \bullet \mathbf{a}_0^n \\ \bullet \mathbf{a}_1^n \\ \cdot \\ \cdot \\ \bullet \mathbf{a}_n^n \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \alpha_n \\ \mathbf{0} \\ \cdot \\ \cdot \\ \mathbf{0} \\ \beta_n \end{bmatrix} \quad \mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{b}_0^n \bullet \\ \mathbf{b}_1^n \bullet \\ \cdot \\ \cdot \\ \mathbf{b}_n^n \bullet \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \delta_n \\ \mathbf{0} \\ \cdot \\ \cdot \\ \mathbf{0} \\ \varepsilon_n \end{bmatrix} \quad (\text{EA.37})$$

La structure de \mathbf{Q} permet de réécrire les relations (EA.37)b et (EA.37)c:

$$\mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{0} \\ \cdot \\ \mathbf{a}_n^{n*} \bullet \\ \cdot \\ \cdot \\ \mathbf{a}_1^{n*} \bullet \\ \mathbf{a}_0^{n*} \bullet \end{bmatrix} = \begin{bmatrix} \beta_n^* \\ \mathbf{0} \\ \cdot \\ \cdot \\ \mathbf{0} \\ \alpha_n^* \end{bmatrix} \quad \mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{0} \\ \cdot \\ \bullet \mathbf{b}_n^{n*} \\ \cdot \\ \cdot \\ \bullet \mathbf{b}_1^{n*} \\ \bullet \mathbf{b}_0^{n*} \end{bmatrix} = \begin{bmatrix} \varepsilon_n^* \\ \mathbf{0} \\ \cdot \\ \cdot \\ \mathbf{0} \\ \delta_n^* \end{bmatrix} \quad (\text{EA.38})$$

¹. La notation $\bullet \mathbf{A}$ signifie que les produits sont effectués avec la matrice \mathbf{A} à droite, alors que $\mathbf{A} \bullet$ signifie que les produits sont effectués avec la matrice \mathbf{A} à gauche. Par défaut (pas de notation pour préciser), c'est le produit matrice à droite qui est utilisé.

Soit deux matrices de dimension (p.p) à priori quelconques, \mathbf{k}_n et \mathbf{l}_n , que l'on utilise pour réaliser des combinaisons entre les relations: ((EA.37)b+(EA.38)b• \mathbf{k}_n) d'une part, ((EA.37)c+ \mathbf{l}_n •(EA.38)a) d'autre part:

$$\begin{aligned}
 \mathbf{Q}_{n+1} \begin{bmatrix} \bullet \mathbf{a}_0^n \\ \bullet (\mathbf{a}_1^n + \mathbf{b}_n^{n*} \cdot \mathbf{k}_n) \\ \vdots \\ \bullet (\mathbf{a}_n^n + \mathbf{b}_1^{n*} \cdot \mathbf{k}_n) \\ \bullet (\mathbf{b}_0^{n*} \cdot \mathbf{k}_n) \end{bmatrix} &= \begin{bmatrix} \alpha_n + \varepsilon_n^* \cdot \mathbf{k}_n \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \beta_n + \delta_n^* \cdot \mathbf{k}_n \end{bmatrix} \\
 \mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{b}_0^n \bullet \\ (\mathbf{b}_1^n + \mathbf{l}_n \cdot \mathbf{a}_n^{n*}) \bullet \\ \vdots \\ (\mathbf{b}_n^n + \mathbf{l}_n \cdot \mathbf{a}_1^{n*}) \bullet \\ (\mathbf{l}_n \cdot \mathbf{a}_0^{n*}) \bullet \end{bmatrix} &= \begin{bmatrix} \delta_n + \mathbf{l}_n \cdot \beta_n^* \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \varepsilon_n + \mathbf{l}_n \cdot \alpha_n^* \end{bmatrix}
 \end{aligned} \tag{EA.39}$$

On veut pouvoir identifier, par un choix approprié de \mathbf{k}_n et \mathbf{l}_n , ces relations aux relations suivantes:

$$\mathbf{Q}_{n+1} \begin{bmatrix} \bullet \mathbf{a}_0^{n+1} \\ \bullet \mathbf{a}_1^{n+1} \\ \vdots \\ \bullet \mathbf{a}_n^{n+1} \\ \bullet \mathbf{a}_{n+1}^{n+1} \end{bmatrix} = \begin{bmatrix} \alpha_{n+1} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \quad \mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{b}_0^{n+1} \bullet \\ \mathbf{b}_1^{n+1} \bullet \\ \vdots \\ \mathbf{b}_n^{n+1} \bullet \\ \mathbf{b}_{n+1}^{n+1} \bullet \end{bmatrix} = \begin{bmatrix} \delta_{n+1} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{EA.40}$$

Pour que (EA.39)a et (EA.39)b d'une part soient identiques à (EA.40)a et (EA.40)b d'autre part, il suffit que:

$$\boxed{\mathbf{k}_n = -\delta_n^{*-1} \cdot \beta_n \quad \mathbf{l}_n = -\varepsilon_n \cdot \alpha_n^{*-1}} \tag{EA.41}$$

ce qui entraîne:

$$\left(\begin{array}{l} \tilde{\mathbf{a}}_0^{n+1} = \tilde{\mathbf{a}}_0^n \\ \tilde{\mathbf{a}}_1^{n+1} = \tilde{\mathbf{a}}_1^n + \tilde{\mathbf{b}}_n^{n*} \cdot \tilde{\mathbf{k}}_n \\ \cdot \\ \cdot \\ \tilde{\mathbf{a}}_n^{n+1} = \tilde{\mathbf{a}}_n^n + \tilde{\mathbf{b}}_1^{n*} \cdot \tilde{\mathbf{k}}_n \\ \tilde{\mathbf{a}}_{n+1}^{n+1} = \tilde{\mathbf{b}}_0^{n*} \cdot \tilde{\mathbf{k}}_n \end{array} \right) \left(\begin{array}{l} \tilde{\mathbf{b}}_0^{n+1} = \tilde{\mathbf{b}}_0^n \\ \tilde{\mathbf{b}}_1^{n+1} = \tilde{\mathbf{b}}_1^n + \tilde{\mathbf{l}}_n \cdot \tilde{\mathbf{a}}_n^{n*} \\ \cdot \\ \cdot \\ \tilde{\mathbf{b}}_n^{n+1} = \tilde{\mathbf{b}}_n^n + \tilde{\mathbf{l}}_n \cdot \tilde{\mathbf{a}}_1^{n*} \\ \tilde{\mathbf{b}}_{n+1}^{n+1} = \tilde{\mathbf{l}}_n \cdot \tilde{\mathbf{a}}_0^{n*} \end{array} \right) \quad (\text{EA.42})$$

ainsi que:

$$\tilde{\alpha}_{n+1} = \tilde{\alpha}_n + \tilde{\varepsilon}_n^{n*} \cdot \tilde{\mathbf{k}}_n \quad \tilde{\delta}_{n+1} = \tilde{\delta}_n + \tilde{\mathbf{l}}_n \cdot \tilde{\beta}_n^{n*} \quad (\text{EA.43})$$

Il reste à calculer les quantités $\tilde{\beta}_{n+1}$ et $\tilde{\varepsilon}_{n+1}$, qui serviront à l'étape suivante de la récursion (afin qu'elles satisfassent aux relations (EA.37)b et (EA.37)c, mais à l'ordre n+1):

$$\tilde{\beta}_{n+1} = \sum_{i=0}^{n+1} \tilde{\mathbf{R}}_{n+2-i} \cdot \tilde{\mathbf{a}}_i^{n+1} \quad \tilde{\varepsilon}_{n+1} = \sum_{i=0}^{n+1} \tilde{\mathbf{b}}_i^{n+1} \cdot \tilde{\mathbf{R}}_{n+2-i} \quad (\text{EA.44})$$

La structure de la matrice \mathbf{Q} permet de réécrire la relation (EA.40)b:

$$\tilde{\mathbf{Q}}_{n+1} \begin{bmatrix} \bullet \tilde{\mathbf{b}}_{n+1}^{n+1*} \\ \bullet \tilde{\mathbf{b}}_n^{n+1*} \\ \cdot \\ \cdot \\ \bullet \tilde{\mathbf{b}}_1^{n+1*} \\ \bullet \tilde{\mathbf{b}}_0^{n+1*} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \tilde{\mathbf{0}} \\ \cdot \\ \cdot \\ \mathbf{0} \\ \tilde{\delta}_{n+1}^{n*} \end{bmatrix} \quad (\text{EA.45})$$

Soit \mathbf{q}_n un vecteur de longueur p. On réalise la combinaison entre les relations ((EA.45)• \mathbf{q}_n)+(EA.37)a):

$$\tilde{\mathbf{Q}}_{n+1} \begin{bmatrix} \bullet (\tilde{\mathbf{b}}_{n+1}^{n+1*} \cdot \mathbf{q}_n + \mathbf{f}_0^n) \\ \bullet (\tilde{\mathbf{b}}_n^{n+1*} \cdot \mathbf{q}_n + \mathbf{f}_1^n) \\ \cdot \\ \cdot \\ \bullet (\tilde{\mathbf{b}}_1^{n+1*} \cdot \mathbf{q}_n + \mathbf{f}_n^n) \\ \bullet (\tilde{\mathbf{b}}_0^{n+1*} \cdot \mathbf{q}_n) \end{bmatrix} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \cdot \\ \cdot \\ \mathbf{g}_n \\ \tilde{\delta}_{n+1}^{n*} \cdot \mathbf{q}_n + \gamma_n \end{bmatrix} \quad (\text{EA.46})$$

Et on veut pouvoir identifier (EA.46) à:

$$\mathbf{Q}_{n+1} \begin{bmatrix} \mathbf{f}_0^{n+1} \\ \mathbf{f}_1^{n+1} \\ \cdot \\ \cdot \\ \mathbf{f}_n^{n+1} \\ \mathbf{f}_{n+1}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{g}_0 \\ \mathbf{g}_1 \\ \cdot \\ \cdot \\ \mathbf{g}_n \\ \mathbf{g}_{n+1} \end{bmatrix} \quad (\text{EA.47})$$

Il suffit pour cela que:

$$\mathbf{q}_n = \underline{\delta}_{n+1}^{*-1} \cdot (\mathbf{g}_{n+1} - \gamma_n) \quad (\text{EA.48})$$

Cela entraîne alors:

$$\begin{cases} \mathbf{f}_0^{n+1} = \mathbf{f}_0^n + \mathbf{b}_{n+1}^{n+1*} \cdot \mathbf{q}_n \\ \mathbf{f}_1^{n+1} = \mathbf{f}_1^n + \mathbf{b}_n^{n+1*} \cdot \mathbf{q}_n \\ \cdot \\ \cdot \\ \mathbf{f}_n^{n+1} = \mathbf{f}_n^n + \mathbf{b}_1^{n+1*} \cdot \mathbf{q}_n \\ \mathbf{f}_{n+1}^{n+1} = \mathbf{b}_0^{n+1*} \cdot \mathbf{q}_n \end{cases} \quad (\text{EA.49})$$

Il reste à calculer la quantité γ_{n+1} , qui servira l'étape suivante de la récursion (afin qu'elle satisfasse à la relation (EA.37)a, mais à l'ordre n+1):

$$\gamma_{n+1} = \sum_{i=0}^{n+1} \underline{\mathbf{R}}_{n+2-i} \cdot \mathbf{f}_i^{n+1} \quad (\text{EA.50})$$

A ce stade, on a achevé l'étape n+1 de la récursion. Il reste donc à initialiser celle-ci (ordre n=0). On vérifie aisément que les initialisations suivantes conviennent:

$$\begin{cases} \mathbf{f}_0^0 = \underline{\mathbf{R}}_0^{-1} \cdot \mathbf{g}_0 \\ \underline{\mathbf{a}}_0^0 = \underline{\mathbf{Id}} & \underline{\mathbf{b}}_0^0 = \underline{\mathbf{Id}} & (\text{Id=Identité}) \\ \underline{\alpha}_0 = \underline{\mathbf{R}}_0 & \underline{\delta}_0 = \underline{\mathbf{R}}_0 \\ \underline{\beta}_0 = \underline{\mathbf{R}}_1 & \underline{\varepsilon}_0 = \underline{\mathbf{R}}_1 \\ \gamma_0 = \underline{\mathbf{R}}_1 \cdot \mathbf{f}_0^0 \end{cases} \quad (\text{EA.51})$$

Ensuite, pour passer de l'ordre n à l'ordre n+1 de la récursion, il suffit d'appliquer successivement les formules: (EA.41), (EA.42), (EA.43), (EA.44), (EA.48), (EA.49), et (EA.50).

Si l'on analyse le temps nécessaire à l'exécution de cet algorithme pour parvenir à l'ordre n :

- Les opérations (EA.41) et (EA.48) nécessitent un temps proportionnel à p^3 (inversions de matrices de dimension p).
- Les opérations (EA.42) et (EA.44) nécessitent un temps proportionnel à $n.p^3$ (n multiplications de matrices de dimension p).
- L'opération (EA.43) nécessite un temps proportionnel à p^3 (une multiplication de matrice de dimension p).
- Les opérations (EA.49) et (EA.50) nécessitent un temps proportionnel à $n.p^2$ (n multiplications matrice-vecteur)

Au total, en tenant compte des n itérations, on parvient à un temps total de l'ordre de (n^2p^3) , à comparer avec $(n.p)^3$ pour une matrice quelconque.

Cas des sous-matrices R_i hermitiennes

Dans le cas où les matrices R_i sont elles-mêmes hermitiennes ($R_i^* = R_i$), les récurrences se simplifient: on vérifie en effet facilement à partir des relations de récurrence (EA.41), (EA.42), (EA.43), (EA.44), (EA.48), (EA.49), et (EA.50) que dans ce cas:

$$\begin{aligned} & \forall i \leq n, \forall j \leq i \\ l_i &= k_i^* & b_j^i &= a_j^{i*} & \delta_i &= \alpha_i^* & \epsilon_i &= \beta_i^* \end{aligned} \tag{EA.52}$$

De sorte que la quantité de calculs nécessaire est environ divisée par 2. En fait toutes les variables introduites à cause de la non-commutativité des produits de matrices deviennent inutiles, les produits étant à nouveau commutatifs comme dans le cas des scalaires.

Annexe D: Abstract SEG Houston'95

Cet abstract (Hugonnet and Canadas, 1995), présenté à la SEG de Houston, traite des questions d'échantillonnage et d'aliasing dans la transformée de Radon parabolique, et reprend donc les éléments du chapitre 2.

Summary

The Parabolic Radon Transform (τ - q) is an usual tool in the framework of long period multiples removing. If the purpose is a stack section without multiples, a parabolic filtering on the CMP gathers after NMO correction is a very efficient process. However, If we are interested in an AVO analysis before stacking, we may observe that the amplitudes of the real events are modified by the process: a part of their energy is aliased in the multiples area of the parabolic decomposition.

A study of the sampling in q and x (offset) for the least-squares decomposition problem is achieved in the frequency domain. Anti-aliasing conditions and the consequences of undersampling are given:

Insufficient q sampling results in the loss of a part of the data.

Insufficient x sampling is responsible for the spreading of the energy in the (τ , q) domain, mainly for high frequencies and far-offset data. To improve the process, this part of the data should be removed. Some other (and more satisfying) ways might be either an improvement of the decomposition itself, or the interpolation of the data in order to create new traces in the far offsets.

Introduction

The use of the Parabolic Radon Transform (PRT) for multiples removal on NMO corrected CMP gathers was first introduced by Hampson (86). Parabolic filtering is very effective, when compared with f - k filtering, in the sense that multiples are uniformly removed at all offsets. Moreover, the primaries should not be modified, so that an AVO study remains possible (Foster and Mosher, 92).

Nevertheless, amplitude anomalies are sometimes observed, and they are due to the spatial undersampling of the data. So we want to study the sampling of the variables involved in the PRT, and understand what happens in case of over- and undersampling.

The comparison of the PRT with the Linear Radon Transform (LRT) is useful, because the PRT can be written as a LRT after an x^2 stretch on the data. The curves used in the transforms are:

Straight lines for the LRT: $t = \tau + p \cdot x$

Parabolas for the PRT: $t = \tau + q \cdot x^2$

Many authors have studied aliasing in the LRT (Turner, 90). Nevertheless, a formal extrapolation to the PRT has to be validated. The interest of the comparison mainly lies in the behavior differences in case of aliasing.

Zhou (94) gave a formulation for LRT and PRT in the frequency domain, in which the least-squares inverse problem is expressed as a deconvolution with respect to plq (we write 'l' to refer to both variables) by a known operator, which depends on the frequency. The problem of sampling can be studied through these operators.

Inverse Problem for the LRT and the PRT

Let us define the PRT (the LRT is very similar): we want to express the data $d(t,x)$ (usually a CMP gather after NMO) as a superposition of constant amplitude parabolas:

$$d(t, x) = \iint_{\tau, q} u(\tau, q) \delta(t - (\tau + qx^2)) d\tau dq \tag{1}$$

This equation is defined to be the forward transform (or modeling) L:

$$d = L \cdot u \tag{2}$$

The coefficients $u(\tau,q)$ are the unknowns of the problem. The least-squares problem associated to equation (2) leads to solve:

$$(L^*L)u = (L^*d) \tag{3}$$

As pointed out by Thorson and Clearbout (85), the adjoint operator L^* is the usual scanning (or stacking) along the parabolas :

$$u_s(\tau, q) = (L^*d)(\tau, q) = \int_x d(\tau + qx^2, x) dx \tag{4}$$

Applying a Fourier transform with respect to time leads to (in Fourier domain, upper cases are used for function names):

$$D(\omega, x) = (LU)(\omega, x) = \int_q U(\omega, q) e^{-i\omega qx^2} dq \tag{5}$$

$$U_s(\omega, q) = (L^*D)(\omega, q) = \int_x D(\omega, x) \cdot e^{i\omega qx^2} \cdot dx \tag{6}$$

From (5) and (6), Zhou (94) showed that L^*L , that is modeling followed by stack, can be written as a convolution with respect to q :

$$U_s(\omega, q) = (L^*LU)(\omega, q) = U(\omega, q) * \sigma_{par}(\omega, q) \tag{7}$$

Where the convolutional operator is :

$$\sigma_{par}(\omega, q) = \int_x e^{i\omega qx^2} \cdot dx \tag{8}$$

For the LRT, the equations are formally the same, replacing q by p and x^2 by x . The operator becomes :

$$\sigma_{lin}(\omega, p) = \int_x e^{i\omega px} \cdot dx \tag{9}$$

Hence, the least-squares inverse can be obtained by : (i) stacking the data to get U_s (equation (6)); (ii) solving for each frequency the deconvolution problem (7). The behavior of the inverse problem is then closely related to the properties of the convolutional operators (8) and (9).

First, we study the operators in the continuous case, with an infinite range of plq , and for $\omega > 0$ (for $\omega < 0$, all the functions turn into their conjugate).

The Fourier transform of a function s with respect to plq is defined by:

$$Fs(k_p) = \int_p s(p) \cdot e^{-ik_p p} \cdot dp$$

The spectrum of σ_{lin} is :

$$F\sigma_{lin}(\omega, k_p) = \iint_{x, p} e^{i(\omega x - k_p)p} \cdot dp \cdot dx = \int_x 2\pi \delta(\omega x - k_p) \cdot dx$$

$$F\sigma_{lin}(\omega, k_p) = \begin{cases} \frac{2\pi}{\omega} & \text{if } \omega x_{min} \leq k_p \leq \omega x_{max} \\ 0 & \text{else} \end{cases} \quad (10)$$

For the PRT, the calculation is virtually identical, except we have to assume that the sign of x is constant (either positive or negative), and change the variable x into: $\chi = \text{sign}(x)\omega x^2$:

$$F\sigma_{par}(\omega, k_q) = \iint_{x_q} e^{i(\omega x^2 - k_q)q} \cdot dq \cdot dx = \iint_{\chi_q} \frac{e^{i(\chi - k_q)q}}{2\sqrt{|\omega|\chi}} \cdot dq \cdot d\chi$$

$$= \int \frac{\pi \delta(|\chi| - k_q)}{\sqrt{|\omega|\chi}} \cdot d\chi$$

$$F\sigma_{par}(\omega, k_q) = \begin{cases} \frac{\pi}{\sqrt{|\omega k_q}} & \text{if } \omega(x^2)_{min} \leq k_q \leq \omega(x^2)_{max} \\ 0 & \text{else} \end{cases} \quad (11)$$

If the sign of x is not constant, we have to split the integral over x into two terms. However, the band in which the spectrum is non-zero is still given by equation (11).

The operators and their spectra are drawn in figure 1.

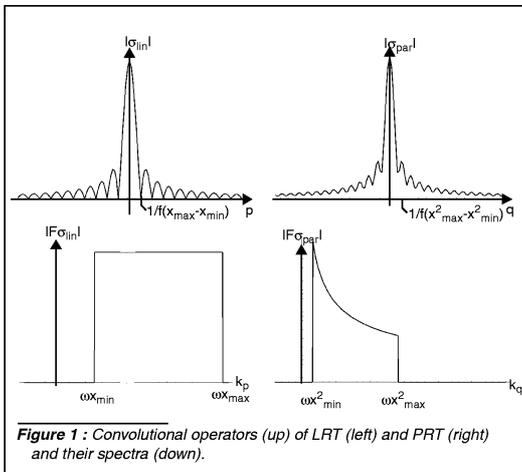


Figure 1 : Convolutional operators (up) of LRT (left) and PRT (right) and their spectra (down).

Sampling in plq

The deconvolution process cannot recover the 'frequencies' (k_p and k_q) cut by the operators. Consequently, and according to the theory of information¹, the critical sampling rates in plq can be derived from equations (10) and (11):

$$\Delta p_c = \frac{1}{f(x_{max} - x_{min})} \quad \Delta q_c = \frac{1}{f((x^2)_{max} - (x^2)_{min})} \quad (12)$$

- Turner (90) gave this result for the LRT, and the PRT really seems to be an extrapolation of the LRT (but for the PRT, negative and positive offsets are undifferentiated).

1. given a function f(x), the sampling rate Δx and the bandwidth B_f of f must satisfy: $\Delta x \cdot B_f < 2\pi$

- When applying the PRT to a CMP gather, $(x^2)_{min}$ is often much smaller than $(x^2)_{max}$ and can be omitted, leading to the relationship given by Kabir (92).
- When $x_{min}=0$, the critical sampling rates correspond to a time-movement of one period ($T=1/f$) between two straight lines or parabolas on the x_{max} trace.

Under- and over-sampling in plq

For the LRT, in case of oversampling in p, not only there is no gain of resolution in (τ, p) space, but also the set of equations becomes ill-conditioned (Kostov, 89) (the spectra of the operators contain large zero bands). The same conclusions hold for the PRT, as illustrated below:

The CMP gather in figure 2 contains only one frequency defining an horizontal event. We observe the residuals (the difference between the original data and their reconstruction from a PRT) as a function of the sampling factor $\alpha = \Delta q / \Delta q_c$ (and q values are chosen in order to avoid the exact value $q=0$).

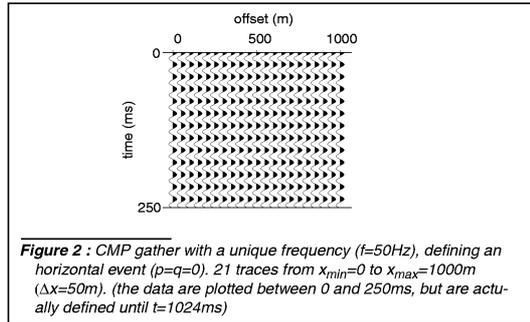


Figure 2 : CMP gather with a unique frequency ($f=50\text{Hz}$), defining an horizontal event ($p=q=0$). 21 traces from $x_{min}=0$ to $x_{max}=1000\text{m}$ ($\Delta x=50\text{m}$). (the data are plotted between 0 and 250ms, but are actually defined until $t=1024\text{ms}$)

PRT's and residuals for several values of α are given in figure 4. From $\alpha=0.9$ (sufficient sampling) to $\alpha=0.3$ (oversampling) the residuals decrease slightly, without any improvement of the resolution in (τ, q) domain. For $\alpha < 1$ all the traces are equally reconstructed, but for $\alpha=1.1$ (undersampling) the far-offset traces are left on the residuals (only these traces are really undersampled: less than 1 parabola per period).

The percentage of residual energy versus α (figure 3) shows a slow decay from $\alpha=0.9$ to 0.1, and a threshold around $\alpha=1$: beyond this limit, more and more traces are left on the residuals.

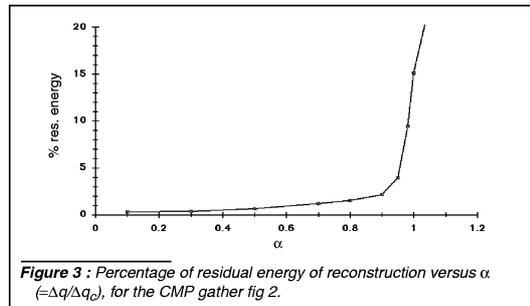


Figure 3 : Percentage of residual energy of reconstruction versus α ($=\Delta q/\Delta q_c$), for the CMP gather fig 2.

We can summarize about the sampling in plq:

- Undersampling in q is equivalent to ignore the far-offset data.

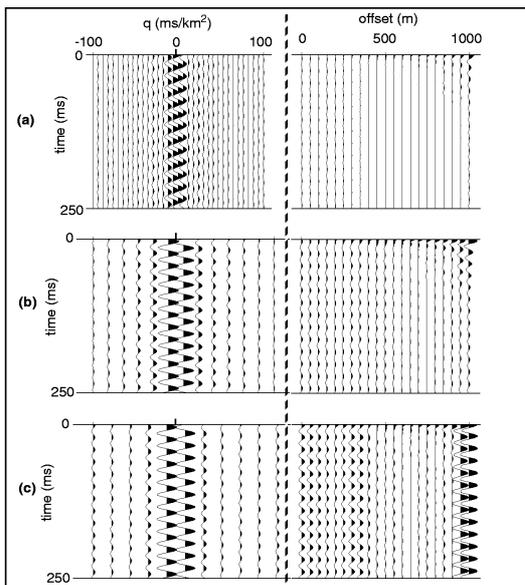


Figure 4 : PRT of CMP gather fig.2 for several values of α ($=\Delta q/\Delta q_c$), and the residuals of the reconstruction (the gain is twice the original data). (a) $\alpha=0.3$; (b) $\alpha=0.9$; (c) $\alpha=1.1$.

- Oversampling results in a slightly better reconstruction, but with numerical problems, an increasing CPU cost, and no gain of resolution.
- In practice, the value $\alpha=0.8$ is a good choice.

Sampling in x

A theoretical study of x sampling is difficult for the PRT, because it involves integrals of the form $\int_{x_{min}}^{x_{max}} e^{i\lambda x^2} dx$, for which there's no analytical solution. In contrast, there is no problem for the LRT: we apply a Fourier transform over x to the modeling equation (5), and then the bandwidth in wavenumber k_x leads to the well known relationship:

$$\Delta x \leq \frac{1}{f(p_{max} - p_{min})} \tag{13}$$

The latter equation can be applied to the PRT, if supposed to be a LRT after the variable transformation $X=x^2$. As $\Delta X = 2x\Delta x$ the condition on ΔX is true for any x if:

$$\Delta x \leq \frac{1}{2f|x|_{max}(q_{max} - q_{min})} \tag{14}$$

However, Δx depends on the data. If it is too large, we want to understand the consequences of the aliasing. We can guess that they are rather different for the LRT and for the PRT, mainly because a regular x sampling induces an irregular (x^2) sampling.

The sampling in x induces important modifications on the convolutional operators, when compared with the continuous case (figure 5) :

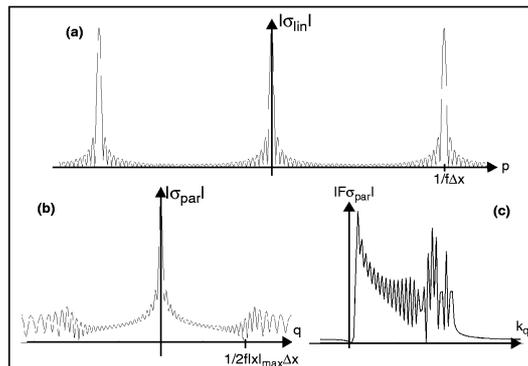


Figure 5 : Secondary lobes appear on operators when x is undersampled. (a) σ_{llr} . (b) σ_{par} . (c) spectrum of σ_{par}

The operator of the LRT becomes periodic, with a period $P=1/f\Delta x$. This results from the impossibility to distinguish two events when their slopes differ from P (Turner, 90): given a real event p^* , each slope $(p^*+j.P)$ (j integer) sums amplitudes in phase (figure 6).

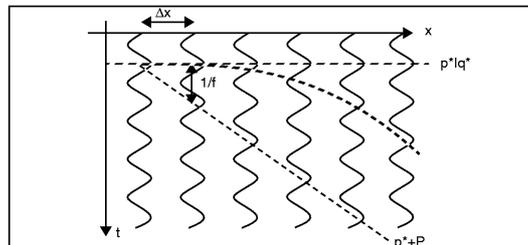


Figure 6 : for a given frequency f, 2 events whose slopes differ from $1/f\Delta x$ are undifferentiated. In PRT, this indetermination does not exist.

When using the PRT, on the contrary, no other parabola than the real one (q^*) sums all amplitudes in phase. Nevertheless there can be local summations in phase: the local slope of a parabola being $(2qx)$, it happens when $|2qx - 2q^*x| = j.P$; so we prevent this if, and only if: $2|x|_{max}|q - q^*| < P$. These summations produce large secondary lobes on the operator σ_{par} (figure 5b), and his spectrum is now very irregular, with large oscillations (figure 5c; $F\sigma_{lin}$ is affected in the same way)

To summarize, we write now as anti-aliasing conditions:

$$|p - p^*| < \frac{1}{f\Delta x} \tag{15}$$

$$|q - q^*| \leq \frac{1}{2f|x|_{max}\Delta x} \tag{16}$$

and we can make the following remarks:

- Aliasing for the LRT is total and well located, while for the PRT it is diffused.
- If the equations (13) and (14) are not verified, the whole deconvolution process is then numerically instable. If the equations (15) and (16) are not verified for a given event $p^*|q^*$, this event is then aliased. Note that instability can arise without aliasing.

Undersampling in x

We use again the CMP gather of figure 2; the only event is at $p^*=q^*=0$. LRT's and PRT's are computed with several values of p_{max} and q_{max} ($p_{min} = -p_{max}$, $q_{min} = -q_{max}$).

Non-aliased and aliased cases are given in figure 7. In case of aliasing, the central event remains unchanged on the LRT, but its amplitude is divided by 3, because there are 2 aliases; on the PRT, it smears on the adjacent traces and its amplitude has also decreased because a part of the energy is located on the secondary lobes.

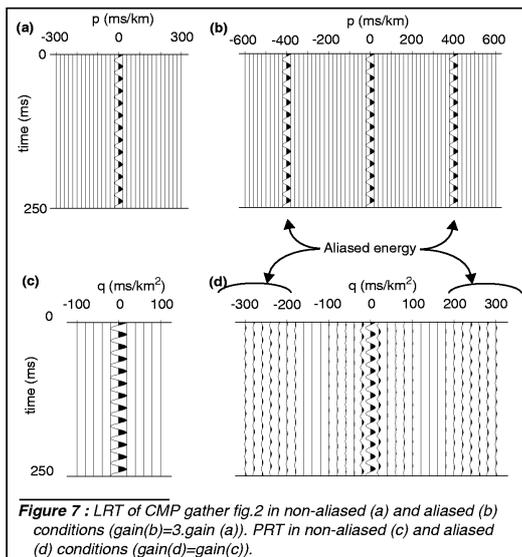


Figure 7 : LRT of CMP gather fig.2 in non-aliased (a) and aliased (b) conditions ($gain(b)=3 \cdot gain(a)$). PRT in non-aliased (c) and aliased (d) conditions ($gain(d)=gain(c)$).

This phenomenon results from a basic property of the 'Least-Squares Inverse': by increasing the plq range, the problem becomes ill-conditioned; there are an infinity of solutions because we allow more possibilities to construct them. Among all these solutions, the process chooses the one having the lowest energy (and not the desired one). Kostov (89) for the LRT, and Darche (90) for the PRT, proposed in case of aliasing to invert LL^* rather than L^*L . The final solution is quite the same, but it prevents numerical instability.

As an example, the two models shown in figure 7a and 7b equally fit the data, but the second one contains 3 events of amplitude 1/3 (energy = $3 \cdot 1/9 = 1/3$), in place of only 1 event of amplitude 1.0 (energy = 1.0).

To summarize, the main consequence of aliasing in x is the spreading of the energy in the (τ, q) space, in place of the expected focusing: in the PRT of a CMP gather, a part of the real events energy is located in the multiples area and vice-versa. It follows that any processing on the ones (modeling and subtraction of the multiples) can affect the others (the real events). The equation (16) shows the ways to prevent this:

- by removing a part of the data, either far offset traces or high frequencies (or both).
- by reducing the q range scanned. But this range is enforced by the events in the data.

Conclusions and suggestions

The study of aliasing in the Parabolic Radon Transform by using the convolution/deconvolution formalism in the frequency domain leads to the determination of critical sampling rates for the variables x and q. More important is the understanding of the behavior of the transform in case of under- and oversampling.

The undersampling in x is in fact the most serious problem, because it depends only on the data (while the sampling in q depends on the user) and it results in the spreading of the energy in the (τ, q) domain. Apart from removing the aliased components of the data (high frequencies on far-off-set traces), we could consider the following solutions:

- To reduce Δx by creating new traces, with an interpolator which reduce spatial aliasing, such as the one proposed by Spitz (91). This should be done only at far offsets, where the undersampling is the strongest.
- A decomposition more parsimonious (Thorson and Claerbout, 85) than the least-squares inverse could prevent the spreading of the energy.

Acknowledgments

We thank Simon Spitz from Compagnie Générale de Géophysique, Eric de Bazelaire and Gilles Darche from Elf Aquitaine Production for many interesting discussions, and Binzhong Zhou from Oxford University for his (τ, plq) code.

References

- DARCHE, G., 1990, Fast parabolic transforms: SEP, **65**, 99-112.
- FOSTER, J., and MOSHER, C., 1992, Suppression of multiple reflections using the Radon transform: Geophysics, **57**, 386-395.
- HAMPSON, D., 1986, Inverse velocity stacking for multiple elimination: Journal of Can. Soc. Expl. Geophys., **22**, 44-55.
- KABIR, M. M. N., 1992, Implementation and applications of the Generalized Radon Transform: M.Sc. Thesis, Delft University.
- KOSTOV, C., 1989, Finite-aperture slant-stack transform: SEP, **61**, 261-279.
- SPITZ, S., 1991, Seismic trace interpolation in the F-X domain: Geophysics, **56**, 785-794.
- THORSON, J. R., and CLAERBOUT, J. F., 1985, Velocity-stack and slant-stack stochastic inversion: Geophysics, **50**, 2727-2741.
- TURNER, G., 1990, Aliasing in the tau-p transform and the removal of spatially aliased coherent noise: Geophysics, **55**, 1496-1503.
- ZHOU, B., and GREENHALGH, A., 1994, Linear and parabolic tau-p transforms revisited: Geophysics, **59**, 1133-1149.

Annexe E: *Abstract SEG Dallas'97*

Cet abstract (Hugonnet and Canadas, 1997), présenté à la SEG de Dallas (1997), traite de rééchantillonnage et d'interpolation par les décompositions de Radon linéaires 3D, et reprend donc les éléments des chapitres 6 et 7.

Regridding of irregular data using 3D Radon Decompositions

Pierre Hugonnet* (1, 2) and Guy Canadas (1); (1): Elf Aquitaine Production; (2): Université de Pau.

SUMMARY

Modern seismic acquisition surveys, especially 3D, often distinguish themselves by irregular geometries and gaps in the traces distribution. On the other hand, many processing steps require regular geometries and no missing traces.

We use 3D Linear Radon Decompositions to regrid a 3D cube along its spatial variables. The classical least-squares solution to the Radon decomposition is useful only for non spatially aliased data. Real data are generally aliased, so we propose to use a sparse Radon decomposition. With this method, results remain accurate for distances between traces up to 6 times the theoretical spatial sampling rates. Larger gaps should be filled using other methods, like prediction filters.

INTRODUCTION

We consider the data $d(t, x, y)$, t being the time variable, (x, y) the two spatial coordinates, and the planes family described by the 3 parameters $(\tau, p_x, p_y) : t = \tau + p_x x + p_y y$

The 3D Linear Radon Decomposition (LRD) is defined in the same way than the 2D one. We want to express $d(t, x, y)$ as a superposition of constant amplitude planes:

$$d(t, x, y) = \mathbf{L}u(t, x, y) = \iint_{p_x, p_y} u(\tau - p_x x - p_y y, p_x, p_y) dp_x dp_y \quad (1)$$

The model $u(\tau, p_x, p_y)$ is the unknown of a linear system. \mathbf{L} is the forward transform (also called reconstruction) from the model space to the data space.

This 3D transform can be decomposed in 2 x 2D (Donati et al., 1996). Consequently, the properties of the 2D transform can be extrapolated to the 3D one.

\mathbf{L}^* -the adjoint of \mathbf{L} - is, as pointed out by Thorson and Claerbout (1985), the usual 'stack operator' of the data along each plane. These authors also showed that the problem is ill-posed (infinity of solutions) for finite aperture data, and proposed several solutions, including the classical least-squares one.

The 2D version of (1) is generally solved in the Fourier domain, because the least-squares inverse can be efficiently obtained by solving a small Toeplitz system for each frequency (Kostov, 1990). In 3D, there's no such Toeplitz structure if the input geometry is irregular, and the Fourier domain brings no more clear advantage. Consequently, we are working in the time domain, which allows to use time-variant weightings and other norms than the least-squares one.

Once we get a Radon Decomposition $u(\tau, p_x, p_y)$ of the data, we can compute the interpolated reconstructed data at any position (x, y) by applying the forward transform (1). The

quality of the interpolation highly depends on the quality of the decomposition.

RADON DECOMPOSITIONS

The decompositions are obtained by minimizing the quantity:

$$\min_u \left(\phi(u) = \frac{1}{2} \|d - \mathbf{L}u\|^2 + \frac{\lambda^2}{2} (u, \mathbf{P}u) \right) \Leftrightarrow u = (\mathbf{L}^* \mathbf{L} + \lambda^2 \mathbf{P})^{-1} \mathbf{L}^* d \quad (2)$$

where $\|\cdot\|$ stands for the L2 norm, and (\cdot, \cdot) for the dot product. \mathbf{P} is a symmetric definite positive operator, which may be a function of u . The first term ensures that the model fits the data, and the second one regularizes the inversion, allowing to choose a particular solution among all the possible ones.

p-sampling

The 2D LRD rules are applied to each variable p_x and p_y . If X and Y are the ranges of x and y , and f_{\max} the maximum frequency of the data, we have to choose (Turner, 1990):

$$\Delta p_x \leq \frac{1}{f_{\max} X} \quad \Delta p_y \leq \frac{1}{f_{\max} Y} \quad (3)$$

Least-Squares (L2-norm) Solution - rho filter

The least-squares solution is obtained with $\mathbf{P} = \mathbf{I}$ (Identity).

As (2) is a quadratic form, it is efficiently solved using a conjugate gradient (CG) method. Lumley (1994) proposed to precondition the gradient with a «rho-filter», whose purpose is to compensate the spectral effects of the transform pair $\mathbf{L}^* \mathbf{L}$.

The 2D LRD pair is a low-pass filter, in $1/|f|$ for infinite apertures (Claerbout, 1985), and the corresponding rho-filter is $|f|$. With finite apertures, low frequencies are actually not amplified up to the infinity: we can show easily that the gain factor for the direct component is $P_x X$, where P_x is the range of p_x . We therefore approximate the actual 2D rho-filter with the following function (a high cut taper is also applied to avoid high frequency noise):

$$\rho_{2D}(X, P_x, f) = \sqrt{f^2 + (1/P_x X)^2} \quad (4)$$

The 3D rho-filter is obtained by multiplying the 2D ones for each direction:

$$\rho_{3D}(f) = \rho_{2D}(X, P_x, f) \rho_{2D}(Y, P_y, f) \quad (5)$$

The gradient derived from (2) is $\mathbf{g}(u) = -\mathbf{L}^*(d - \mathbf{L}u) + \epsilon \mathbf{P}u$. The preconditioned gradient is: $\mathbf{g}_1(u) = \mathbf{R} \mathbf{g}(u)$, where \mathbf{R} is

3D Radon Regridding

the rho-filter operator. The convergence is then reached in only 1 or 2 iterations for regular geometries and about 3 to 5 iterations for fully random geometries.

Lumley found that the rho-filter was speeding up the convergence at the beginning, but that the CG was diverging after a few iterations. Nothing like that has been noticed here: the rho-filtered convergence is much better, even after many iterations.

Sparse Solutions

The sparse Radon Decompositions have been introduced by Thorson and Claerbout (1985). Their method, based on minimum entropy, leads to a model with the less possible non-zero samples. Nichols (1994) presented a solution by minimizing the L1-norm of the model instead of the L2-norm. The solution of Sacchi and Ulrych (1995) is close to the one of Thorson and Claerbout. All of them are sparse solutions.

With sparse solutions, a constant amplitude plane is mapped in the model space into a single point. This highly desirable property is not obtained with classical solutions, because of the finite aperture of the data. Another interesting property is the spatial aliasing attenuation: spatial aliasing results in the spreading of the energy at multiple locations in the model space (Turner, 1990; Hugonnet and Canadas, 1995), which is inconsistent with the sparseness (concentration of the energy into a few non-zero samples).

The disadvantages are: (i) a poor representation of events which do not match the model (curved lines, varying amplitudes, ...): they cannot be mapped into single points and may be classed as modeling noise; (ii) a higher CPU cost. These two problems can be more or less solved by dividing the input data into small patches, on which events may be approximated by constant amplitude planes, and processing each patch independently from the other ones.

We use here a «L1-norm style» solution: \mathbf{P} in equ. (2) is a diagonal weight defined as follows:

$$\mathbf{P} = \frac{1}{\bar{u} + \epsilon} \quad \bar{u} = \frac{1}{T} \int_{t-\frac{T}{2}}^{t+\frac{T}{2}} |u| dt \quad (6)$$

\bar{u} is a smoothed version of $|u|$ over the length T of the seismic wavelet, and ϵ is a damping factor (generally: $\epsilon = \bar{u}_{max}/100$).

As \mathbf{P} depends on \mathbf{u} , (2) must be solved using an IRLS (Iteratively Reweighted Least-Squares) algorithm, where \mathbf{P} is updated from time to time during the iterative CG process (Nichols, 1994). Our approach is slightly different:

- (i) compute an approximate L2-norm solution
- (ii) compute the weight \mathbf{P} from the last known solution.
- (iii) solve the problem (2), keeping the weight constant.
- (iv) stop, or goto (ii)

In the CG iterations of the step (iii), the gradient is preconditioned by the inverse of the weight: $\mathbf{g}_j(\mathbf{u}) = \mathbf{P}^{-1} \mathbf{g}(\mathbf{u})$.

At the end of each cycle (step (iv)), we have a 3D LRD of \mathbf{d} , in the sense that the model fits the data. Each new cycle gives a solution closer to the L1-norm solution. In practice we stop the

process at the end of the second cycle: the solution is very close to the real L1-norm solution, and it requires about one half of the iterations needed with the usual IRLS algorithm.

REGRIDDING

We apply the method to synthetic 2D data: the 2 coordinates of the problem are (x_r, x_s) : shot and receiver locations. The reference is a regular survey SR50 where $\Delta x_s = \Delta x_r = 50m$, $2000 \leq x_s \leq 3950$, $x_s - 2000 \leq x_r \leq x_s - 50$. Two irregular surveys, SI20 and SI50, have been simulated, with randomly located shots and receivers. These surveys are defined by their mean intertrace and intershot: For SI20: $\overline{\Delta x_s} = \overline{\Delta x_r} = 20m$; for SI50: $\overline{\Delta x_s} = \overline{\Delta x_r} = 50m$. Moreover, traces have been removed from a rectangular area ($1350 < x_r < 1650$; $2350 < x_s < 2650$), see fig. 1. Two CSPs from SR50 are plotted on fig. 2.

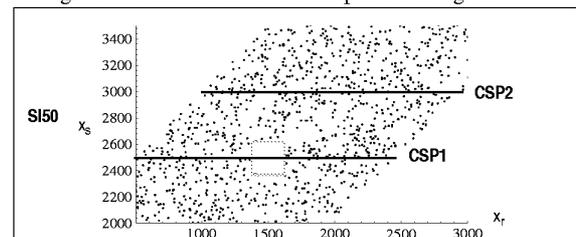


Figure 1 : Survey SI50. (x_r, x_s) are the receiver and source x-position. Mean intertrace and intershot are equal to 50m. Traces in the grey box have been removed. Continuous lines are the 2 two studied CSPs. SI20 is a similar survey with mean intertrace and intershot equal to 20m.

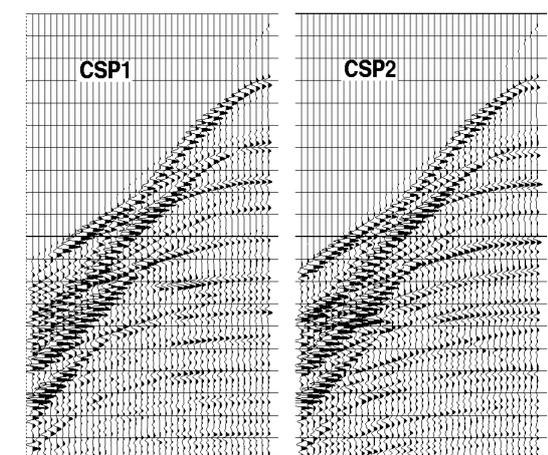


Figure 2 : 2 CSPs from the reference SR50, at $x_s=2500m$ and $x_s=3000m$

Flex-Binning

The easiest way to regrid an irregular survey is the flex-binning: for each bin (50x50m) we choose the closer trace to the center of bin within the bin itself and its immediate neighbors. The fig. 2 shows the flex-binning of SI50 (only the residuals - difference between the binning and the exact solution given by

3D Radon Regridding

SR50- are plotted): as expected the binning works fine for horizontal events, but is not valid for dipping ones.

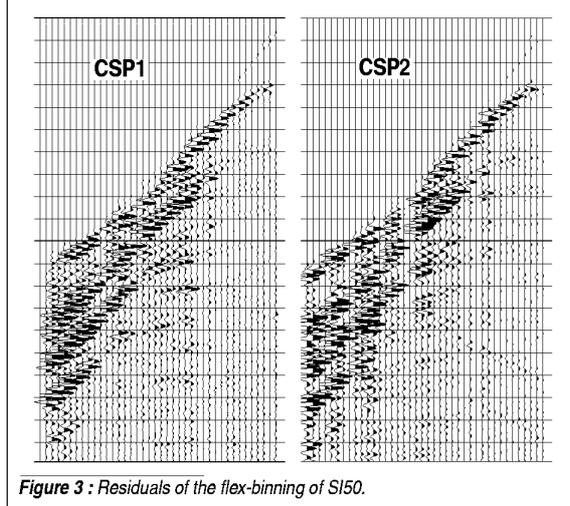


Figure 3 : Residuals of the flex-binning of SI50.

L2-norm Radon Regridding

The 3D LRDs are computed with the following parameters: $-0.7 \leq p_{xt} \leq 0.3 \text{ms/m}$, $-0.3 \leq p_{xs} \leq 0.7 \text{ms/m}$ (the domains are asymmetric because we have only negative offsets). As $f_{max} = 50 \text{Hz}$, the spatial aliasing conditions are (Turner, 1990):

$$\begin{aligned} \Delta x_s &\leq (\Delta x_s)_{th} = 1 / (f_{max} P_{xs}) = 20 \text{m} \\ \Delta x_r &\leq (\Delta x_r)_{th} = 1 / (f_{max} P_{xr}) = 20 \text{m} \end{aligned} \quad (7)$$

The size of the patches is 250x250m.

Equ. (7) means that SI50 is spatially aliased (in average), but not SI20. As a consequence, the L2-norm Radon regridding of SI20 is much better than the regridding of SI50, except where there are some gaps in the traces distribution (fig. 4).

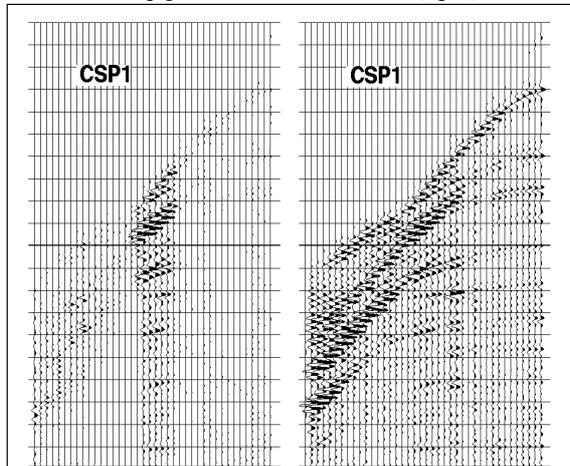


Figure 4 : residuals of the L2-norm Radon regridding of SI20 (left) and SI50 (right).

L1-norm Radon Regridding

The use of a sparse 3D LRD clearly improves the regridding of SI50 (fig. 5).

Nevertheless, the interpolation still fails in areas where there are not enough traces. We therefore need a criterium to keep or

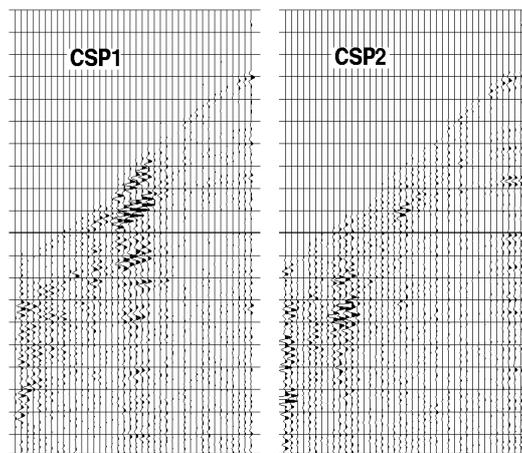


Figure 5 : Residuals of the L1-norm Radon regridding of SI50.

reject interpolated traces: around each interpolated trace, we consider the data traces within a radius R_v : the angular distance between two consecutive traces must not exceed π (fig. 6). The choice of R_v is not obvious (if too large, bad traces are kept, if too small, too much traces are rejected), it should be chosen as a function of the theoretical spatial sampling rates given by (7).

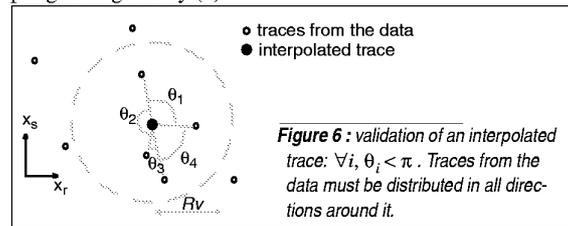


Figure 6 : validation of an interpolated trace: $\forall i, \theta_i < \pi$. Traces from the data must be distributed in all directions around it.

A good choice seems to be: $R_v = 3(\Delta x_s)_{th} = 60 \text{m}$. The L1-norm regridding of SI50 after the rejection is on fig. 5: all the kept traces have low residuals.

Anyway, the rectangular gap on CSP1 shows that it is difficult to fill large gaps with Radon techniques. With the L1-norm, the maximum interpolation distance is around 3 times the theoretical spatial sampling rates (this is a radius, which corresponds to a distance between data traces up to 6 times). As these rates depend on the used p-ranges (7), the choice of these latter is very important: they should be chosen as small as possible, perhaps varying over the data (different for positive and negative offsets for example).

Effects on further processing steps

We illustrate the effects of the regridding on further processing steps: A NMO+stack of the survey SI20 is performed after a flex-binning (on a 25x25m grid) and after a L2-norm Radon

3D Radon Regridding

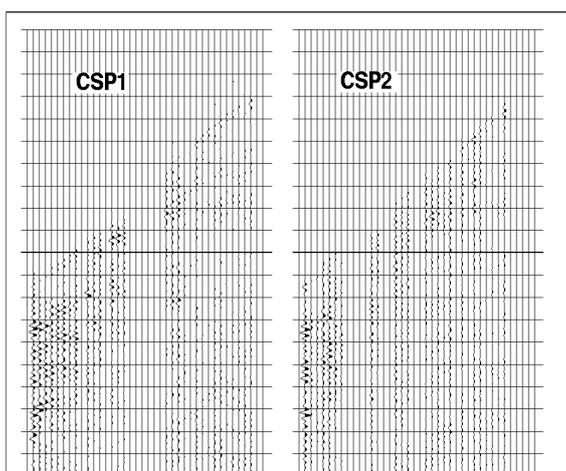


Figure 7 : Residuals of the L1-norm Radon regridding of SI50 with rejection of badly interpolated traces (compare to fig. 5).

regridding. The stack traces are located between $x=2000\text{m}$ and 3000m . The positioning errors of the binned data induce noise on the stack, and dipping reflectors are more or less attenuated (fig. 8).

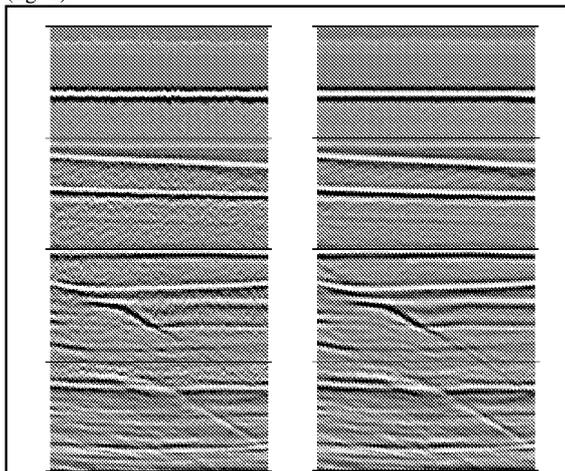


Figure 8 : stack sections from the survey SI20. Left: after a flex-binning. right: after a L2-norm Radon regridding.

This example would of course not arise in the reality: this is a basically a 2D surveys, in which the variables (x_r, x_s) could be reduced to only one (mid-point x -coordinate) after NMO. We have chosen here to artificially keep the two variables and work before NMO, in order to demonstrate the method. With a 3D survey, the 4 spatial variables would reduce to 2 after NMO, on which a 3D LRD could be performed.

Nevertheless, the results can be extrapolated to any processing step which does not account for the real coordinates of the traces.

CONCLUSIONS

We have shown that 3D Radon techniques are efficient to regrid irregular 3D data geometries. Real data are generally spatially aliased (even if they aren't in average, irregularity induces local aliasing), and sparse decompositions are then needed. It is possible with them to interpolate across gaps up to 6 times the theoretical sampling rates. For larger gaps, prediction filters techniques (Huard and Spitz, 1997), seem to be more effective.

Many processing steps (including prediction filters) require regularly sampled data, and some of them are much more efficient if the data are not aliased: a previous regridding can then improve their results. From another point of view, it is easier to design and use algorithms for regular geometries rather than for irregular ones: a regridding at the beginning of the processing sequence could save time for the remaining sequence.

ACKNOWLEDGEMENTS

This work is part of a COPREP project CGG/ELF. We thank Irene Huard and Simon Spitz from CGG for their contributions and interesting discussions.

REFERENCES

- Claerbout, J. F., 1985, Imaging the earth's interior: Blackwell Scientific Publications.
- Donati, M. S., Martin, N. W., and Stewart, R. R., 1996, 3D tau-p filtering: 58th Mtg. Eur. Assoc. Expl. Geoph., Extended Abstracts, Session B004.
- Huard, I., and Spitz, S., 1997, An application of 3D filtering to the restoration of missing traces: 59th Mtg. Eur. Assoc. Expl. Geoph., Extended Abstracts.
- Hugonnet, P., and Canadas, G., 1995, Aliasing in the parabolic radon transform: 65th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1366-1369.
- Kostov, C., 1990, Toeplitz structure in slant-stack inversion: 60th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1618-1621.
- Lumley, D. E., 1994, Estimating a pseudo-unitary operator for velocity stack inversion: SEP report, **82**, 63-78.
- Nichols, D., 1994, Velocity-Stack inversion using L_p norms: SEP report, **82**, 1-16.
- Sacchi, M. D., and Ulrych, T. J., 1995, Model re-weighted least-squares Radon operators: 65th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 616-618.
- Thorson, J. R., and Claerbout, J. F., 1985, Velocity stack and slant stack stochastic inversion: Geophysics, **50**, 2727-2741.
- Turner, G., 1990, Aliasing in the tau-p transform and the removal of spatial aliased coherent noise: Geophysics, **55**, 1496-1503.

Annexe F: Poster Vail'97

Ce poster, qui reprend le même sujet que le papier présenté à la SEG de Dallas (Hugonnet and Canadas, 1997), a été présenté pendant le '1997 SEG summer research workshop', du 3 au 8 Août 1997 à Vail (Colorado), dont le sujet était: 'The interaction between seismic acquisition, processing, and interpretation: understanding and optimizing cost, quality, and turnaround.'

Regridding of irregular data using 3D Radon Decompositions

Pierre Hugonnet* (1, 2) and Guy Canadas (1); (1): Elf Aquitaine Production; (2): Université de Pau.

We use 3D Linear Radon Decompositions to regrid a 3D cube along its spatial variables. The classical least-squares solution to the Radon decomposition is useful only for non spatially aliased data. Real data are generally aliased, so we propose to use a sparse Radon decomposition. With this method, results remain accurate for distances between traces up to 6 times the theoretical spatial sampling rates. Larger gaps should be filled using other methods, like prediction filters.

RADON DECOMPOSITIONS

The 3D Linear Radon Decomposition (LRD) is defined in the same way than the 2D one. We want to express $d(t, x, y)$ as a super-

position of constant amplitude planes: $d(t, x, y) = \mathcal{L}u(t, x, y) = \int \int_{p_x, p_y} u(t - p_x x - p_y y, p_x, p_y) dp_x dp_y$

\mathcal{L} is the forward transform (also called reconstruction) from the model space to the data space. Once we get a Radon Decomposition $u(\tau, p_x, p_y)$ of the data, we can compute the interpolated reconstructed data at any position (x, y) by applying the forward transform. The quality of the interpolation highly depends on the quality of the decomposition.

The decompositions are obtained by minimizing the quantity: $\min_u \left(\phi(u) = \frac{1}{2} \|d - \mathcal{L}u\|^2 + \frac{\lambda^2}{2} (u, \mathcal{P}u) \right) \Leftrightarrow u = (\mathcal{L}^* \mathcal{L} + \lambda^2 \mathcal{P})^{-1} \mathcal{L}^* d$.
 $\|\cdot\|$ stands for the L2 norm, (\cdot, \cdot) for the dot product. \mathcal{P} is a symmetric definite positive operator, which may be a function of u .

The least-squares solution is obtained with $\mathcal{P} = \mathcal{I}$ (Identity).

The sparse Radon Decompositions have been introduced by Thorson and Claerbout (1985). Their method, based on minimum entropy, leads to a model with the less possible non-zero samples, which is equivalent to assume a limited number of independent events. Nichols (1994) presented a solution by minimizing the L1-norm of the model instead of the L2-norm. The solution of Sacchi and Ulrych (1995) is close to the one of Thorson and Claerbout. All of them are sparse solutions.

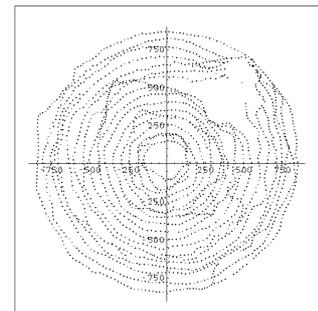
With sparse solutions, a constant amplitude plane is mapped in the model space into a single point. This highly desirable property is not obtained with classical solutions, because of the finite aperture of the data. Another interesting property is the spatial aliasing attenuation: spatial aliasing results in the spreading of the energy at multiple locations in the model space, which is inconsistent with the sparseness (concentration of the energy into a few non-zero samples).

The disadvantages are: (i) a poor representation of events which do not match the model (curved ones, varying amplitudes, ...): they cannot be mapped into single points and may be classed as modeling noise; (ii) a higher CPU cost. These problems can be more or less solved by dividing the data into small patches, on which events may be approximated by constant amplitude planes, and processing each patch independently from the other ones.

We use here a «L1-norm style» solution: \mathcal{P} is a diagonal weight defined as follows: $\mathcal{P} = \frac{1}{\bar{u} + \epsilon}$ $\bar{u} = \frac{1}{T} \int_{-T/2}^{T/2} |u| dt$. This weight is almost constant over the signal length T, but simulates the L1 norm of the model at a larger scale.

REGRIDDING

The process (Radon Decomposition + Reconstruction) can be applied to any data cube with 2 spatial variables. One application is the regridding of a marine 3D VSP, whose shots are located on concentric circles around the receivers position, leading to a fully irregular traces distribution (fig. on the right). The regridding is performed on constant receiver collections. For 3D surface seismic, the problem is the choice of a domain (constant offset, constant shot, constant line...): perhaps should we think about 4D or even 5D Radon Decompositions (there shouldn't be any theoretical problem for this).



The use of sparse decompositions clearly improves the quality of the regridding (in fact classical least-squares decompositions are generally not suited to real data, because of the spatial aliasing). It allows to interpolate across gaps up to 6 times the theoretical spatial sampling rates given by Turner (1990). As these rates depend on the p-ranges used in the decomposition, the choice of these latter is very important: they should be chosen as small as possible, perhaps varying over the data. Larger gaps should be filled with other methods, like prediction filters (Huard and Spitz, 1997).

The regridding should improve the quality of all processing steps which are designed to work with regular and non aliased input data (they are numerous, and real data are most of time irregular and aliased).

REFERENCES

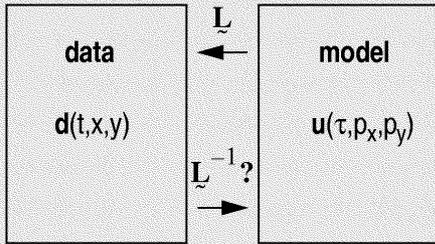
- Huard, I., and Spitz, S., 1997, An application of 3D filtering to the restoration of missing traces: 59th Mtg. Eur. Assoc. Expl. Geoph., Extended Abstracts.
- Nichols, D., 1994, Velocity-Stack inversion using Lp norms: SEP report, **82**, 1-16.
- Sacchi, M. D., and Ulrych, T. J., 1995, Model re-weighted least-squares Radon operators: 65th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 616-618.
- Thorson, J. R., and Claerbout, J. F., 1985, Velocity stack and slant stack stochastic inversion: Geophysics, **50**, 2727-2741.
- Turner, G., 1990, Aliasing in the tau-p transform and the removal of spatial aliased coherent noise: Geophysics, **55**, 1496-1503.



Regridding of irregular geometries using 3D Radon Decompositions

Pierre Hugonnet (1, 2) Guy Canadas (1) 1: Elf EP, 2: Université de Pau (France)

Radon Decompositions - Inverse Problem



Forward Transform: superposition of constant amplitude planes

$$d(t, x, y) = \mathcal{L}u(t, x, y) = \int \int \int_{\tau, \rho_x, \rho_y} u(\tau, \rho_x, \rho_y) \delta(t - (\tau + \rho_x x + \rho_y y)) d\tau d\rho_x d\rho_y$$

Inverse Transform (decomposition):
Ill-posed problem (infinity of solutions)

$$\min_u \left(\phi(u) = \frac{1}{2} \|d - \mathcal{L}u\|_{L_2}^2 + \frac{\lambda^2}{2} (u, \mathcal{P}u) \right)$$

\Leftrightarrow

$$u = (\mathcal{L}^* \mathcal{L} + \lambda^2 \mathcal{P})^{-1} \mathcal{L}^* d$$

Reconstruction: applying \mathcal{L} to a decomposition u , at any position (x, y) .

L2 norm decomposition

$$\mathcal{P} = \mathcal{I}_d$$

Sparse decompositions

Map the data into the less possible significant samples of the model

Stochastic inverse (Thorson and Claerbout, 1985) $\mathcal{P}_{ii} = \sigma_n^2 / u_i^2$

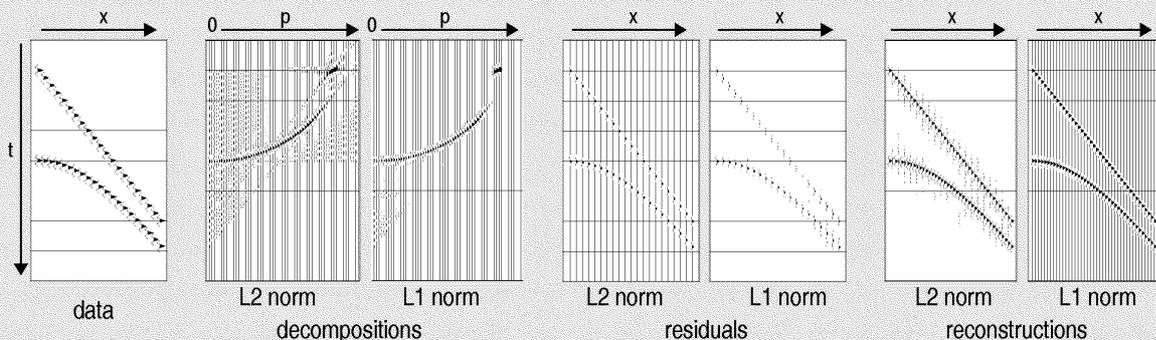
L1 norm minimization (Nichols, 1994) $\mathcal{P}_{ii} = 1 / (|u_i| + \epsilon)$

...(Sacchi and Ulrych, 1995)

Conjugate Gradient in time domain
Rho-filter (ω^2) preconditionning

Irregularity => more iterations

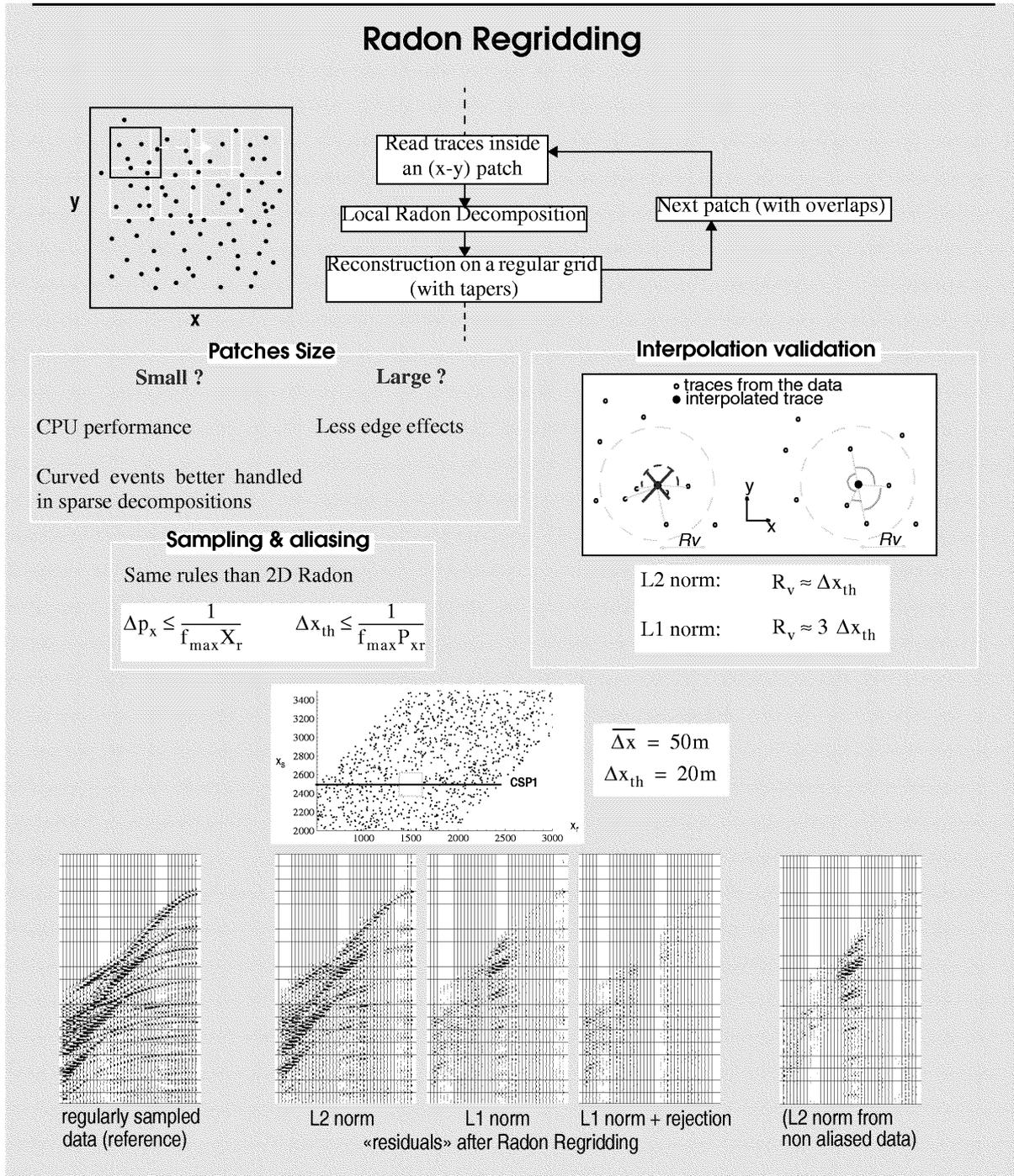
IRLS algorithm based on Conjugate Gradient





Regridding of irregular geometries using 3D Radon Decompositions

Pierre Hugonnet ^(1, 2) Guy Canadas ⁽¹⁾ 1: Elf EP, 2: Université de Pau (France)



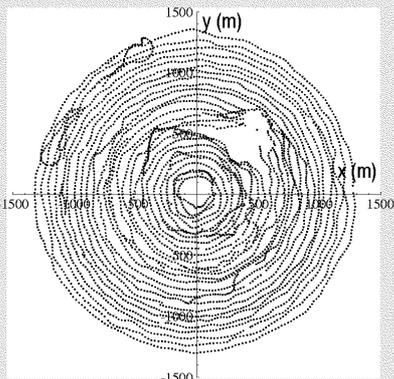


Regridding of irregular geometries using 3D Radon Decompositions

Pierre Hugonnet ^(1, 2) Guy Canadas ⁽¹⁾ 1: Elf EP, 2: Université de Pau (France)

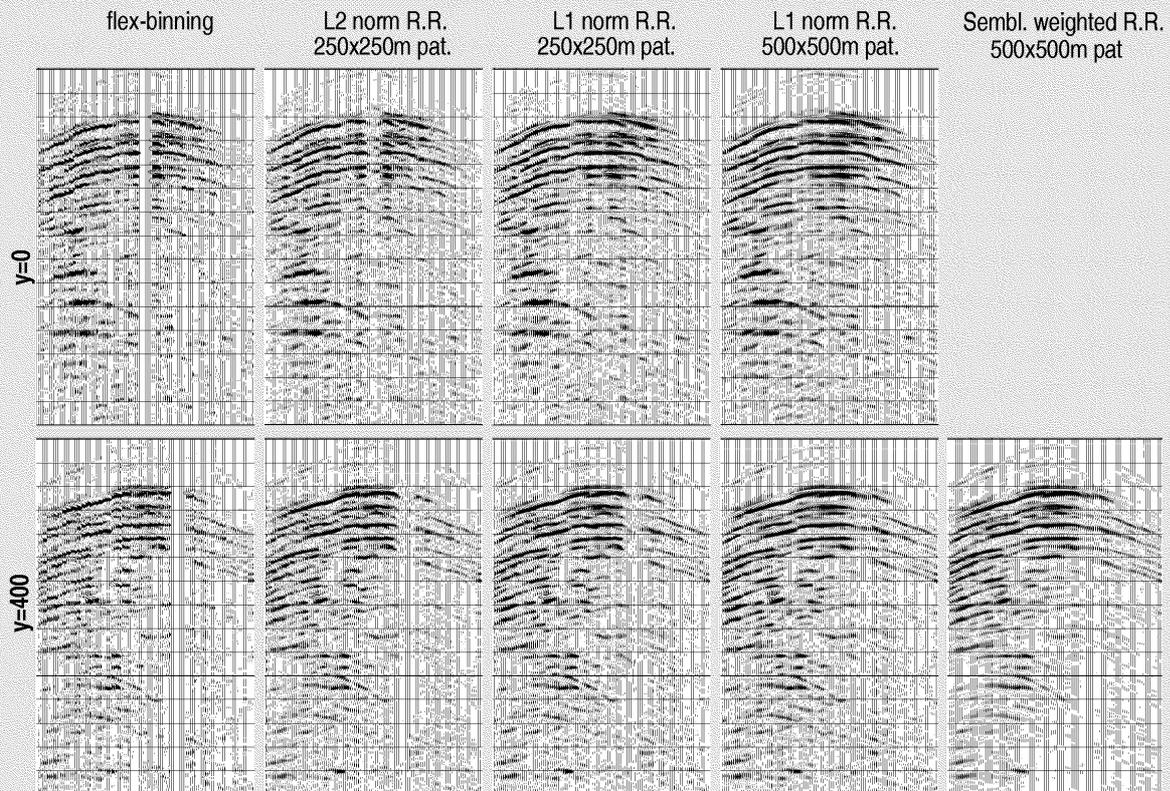
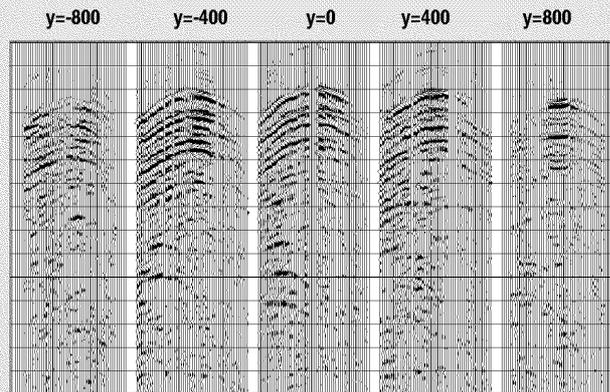
Application: Radon Regridding of a 3D VSP

Concentric circles acquisition around the receivers location
 InterShot = 25m ; InterCircle = 50m



Sampling requirements: $\Delta x = \Delta y = 25m$

Extracted Upgoing P-wave field





Regriding of irregular geometries using 3D Radon Decompositions

Pierre Hugonnet ^(1, 2) Guy Canadas ⁽¹⁾ 1: Elf EP, 2: Université de Pau (France)

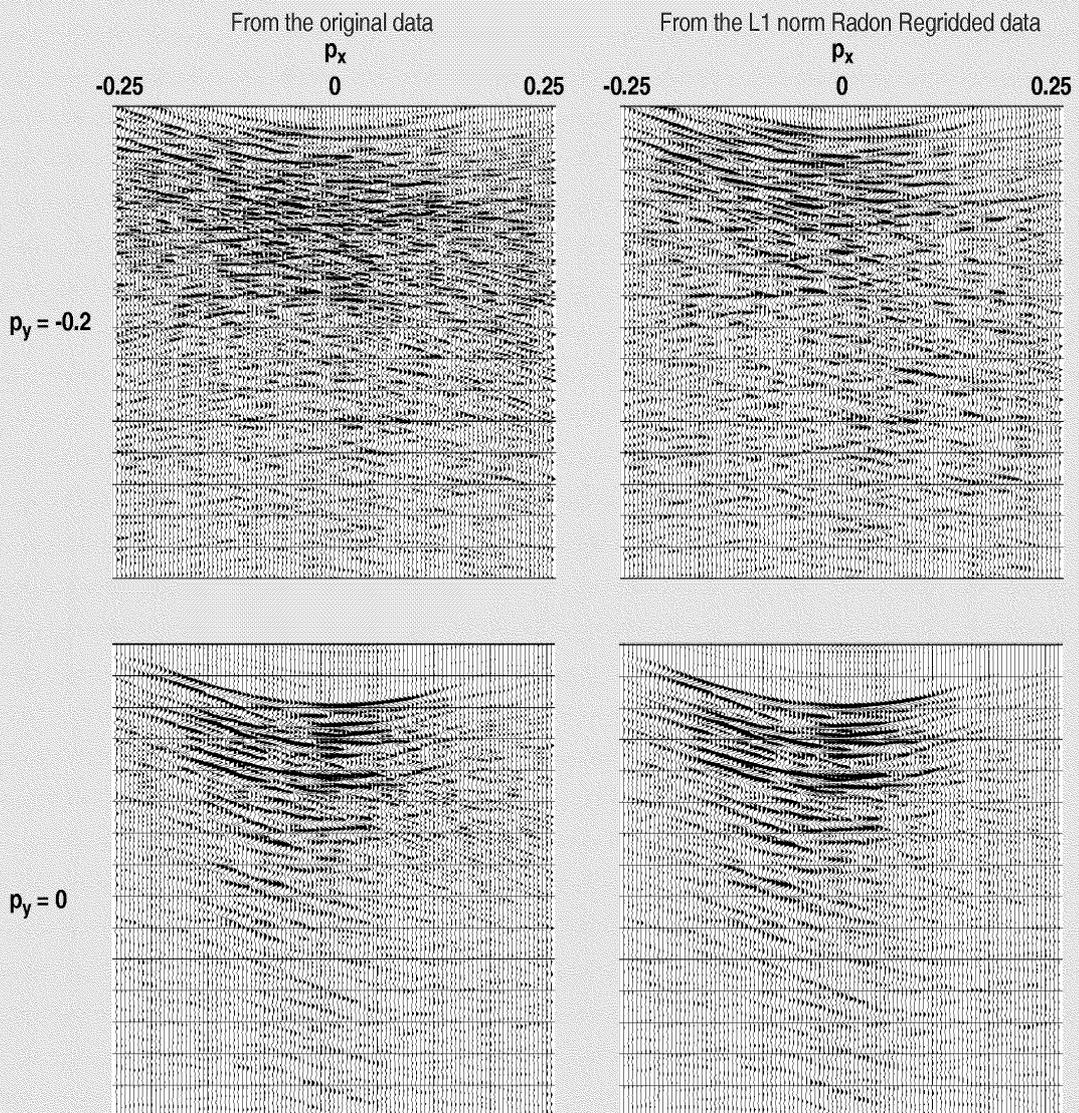
Application: 3D VSP

effect of R.R. on further processing steps

Global Plane-wave decomposition of the 3D VSP

$$-0.25 \leq p_x \leq 0.25 \text{ ms/m}$$

$$-0.25 \leq p_y \leq 0.25 \text{ ms/m}$$



Nomenclature

N.1 - Glossaire

BF	Basse Fréquence
CMP	Common Mid-Point (collection de traces en point milieu commun)
DRH	Décomposition de Radon Hyperbolique
DRL	Décomposition de Radon Linéaire
DRP	Décomposition de Radon Parabolique
(F)FT	(Fast) Fourier Transform
GC	Gradient Conjugué
HF	Haute fréquence
HRD / HRT	Hyperbolic Radon Decomposition/Transform
IRLS	Iteratively Reweighted Least-Squares
LRD / LRT	Linear Radon Decomposition/Transform
pdf	Probability Density Function
PRD / PRT	Parabolic Radon Decomposition/Transform
TF(R)	Transformée de Fourier (Rapide)
TRH	Transformée de Radon Hyperbolique
TRL	Transformée de Radon Linéaire
TRP	Transformée de Radon Parabolique

N.2 - Divers

N.2.1 - scalaires, vecteurs, matrices

Dans les équations, les scalaires sont notés en écriture normale, les vecteurs en gras, et les matrices en gras soulignées par un tilde:

Scalaire: a . Vecteur: vecteur: \mathbf{x} . Matrice: $\tilde{\mathbf{L}}$

La matrice **transposée** s'écrit $\tilde{\mathbf{L}}^T$

La matrice **hermitienne** (transposée+conjuguée) s'écrit: $\tilde{\mathbf{L}}^H$

Le **produit scalaire** de deux vecteurs \mathbf{x} et \mathbf{y} est noté: (\mathbf{x}, \mathbf{y}) , ou $\mathbf{x}^t \mathbf{y}$, ou $\mathbf{x} \cdot \mathbf{y}$. Ces deux dernières notations sont réservées au produit scalaire euclidien.

N.2.2 - Opérateurs

La notation choisie pour les opérateurs est la même que pour les matrices, permettant d'avoir la même écriture pour les problèmes continus et discrets: $\tilde{\mathbf{L}}$.

L'opérateur **adjoint** est noté: $\tilde{\mathbf{L}}^*$ (il est défini par la relation entre produits scalaire pour tous \mathbf{d} et \mathbf{u} : $(\mathbf{d}, \tilde{\mathbf{L}}\mathbf{u}) = (\tilde{\mathbf{L}}^*\mathbf{d}, \mathbf{u})$)

N.3 - Transformées de Fourier

On utilise les définitions suivantes pour la Transformée de Fourier (T.F.) de fonctions:

Soit $f(x)$ une fonction, réelle ou complexe, de la variable réelle x . Sa transformée de Fourier -ou analyse Fourier-, notée Ff ou \hat{f} , est définie par:

$$Ff(k_x) = \hat{f}(k_x) = \int_x f(x) e^{-ixk_x} dx \quad (EN.1)$$

La transformée inverse -ou synthèse de Fourier- d'une fonction $g(k_x)$ est:

$$F^{-1}g(x) = \frac{1}{2\pi} \int_{k_x} g(k_x) e^{ixk_x} dx \quad (EN.2)$$

Transformées de Fourier sur plusieurs variables

si \mathbf{x} est un vecteur: $\mathbf{x} = (x_1, x_2, \dots, x_n)$. La TF à n dimensions de $f(\mathbf{x})$ est la fonction:

$$F_{\mathbf{x}}f(\mathbf{k}_{\mathbf{x}}) = \int_{\mathbf{x}} f(\mathbf{x}) e^{-i(\mathbf{x}^t \mathbf{k}_{\mathbf{x}})} d\mathbf{x} \quad (EN.3)$$

Où $\mathbf{k}_{\mathbf{x}}$ est un vecteur de même dimension que \mathbf{x} : $\mathbf{k}_{\mathbf{x}} = (k_{x_1}, k_{x_2}, \dots, k_{x_n})$

N.4 - Distributions

Pour manipuler les distributions, on utilise les notations suivantes, extraites de Roddier (1971):

On considère ici fonctions et des distributions réelles ou complexes, d'une ou plusieurs variables réelles ou complexes. Dans les notations les variables peuvent donc être des scalaires aussi bien que des vecteurs.

Les intégrales seront prises au sens de **Lebesgues**.

Si E est l'espace sur lequel sont définies les fonctions et distributions (on ne précisera pas à chaque fois fonction *sur E* ou distribution *sur E*)

$L_\alpha(E)$ est l'espace des fonctions f sur E telles que $\|f\|_{L_\alpha} < +\infty$; $\left(\|f\|_{L_\alpha} = \left[\int_E |f|^\alpha \right]^{1/\alpha} \right)$. Si $\alpha \geq 1$,

$L_\alpha(E)$ est un espace vectoriel normé.

$D_0(E)$ est l'espace vectoriel des fonctions sur E *continues à support borné*.

$D(E)$ est l'espace vectoriel des fonctions *indéfiniment dérivables à support borné*.

$S(E)$ est l'espace vectoriel des fonctions de *variable réelle, indéfiniment dérivables, décroissant vers l'infini*, ainsi que chacune de leurs dérivées, *plus rapidement que toute puissance de 1/|x|* (ces fonctions, ainsi que leurs dérivées, sont donc bornées et sommables).

On appelle fonctionnelle sur X (X étant un ensemble de fonctions) un opérateur f qui à toute fonction de X associe un scalaire. La notation standard est:

$$\begin{aligned} X &\rightarrow \mathbb{IR} \text{ (ou } \mathbb{C}) \\ \varphi &\mapsto \langle f, \varphi \rangle \end{aligned} \quad (\text{EN.4})$$

On appelle **distribution** sur E toute fonctionnelle linéaire continue sur $D(E)$; $D'(E)$ est l'espace vectoriel des distributions sur E (D' est appelé l'espace dual de D).

On appelle **distribution régulière** une distribution f telle que il existe une fonction f(x) localement sommable (sommable sur tout ensemble borné) telle que:

$$\langle f, \varphi \rangle = \int_x f(x)\varphi(x)dx \quad (\text{EN.5})$$

Dans les notations, on confond en général la distribution f, et la fonction f(x).

On appelle **distribution tempérée** sur E toute fonctionnelle linéaire continue sur $S(E)$; $S'(E)$ est l'espace vectoriel des distributions tempérées sur E.

La **transformée de Fourier FT** d'une distribution tempérée T est la distribution tempérée définie par:

$$\langle FT, \varphi \rangle = \langle T, F\varphi \rangle \quad (\text{EN.6})$$

(cette définition est possible car la T.F. d'une fonction de $S(E)$ est une fonction de $S(FE)$, FE étant le domaine image de E par la T.F.; cela n'est pas vrai avec les fonctions de D , ce qui empêche de définir la T.F. d'une distribution quelconque).

N.5 - Transformée de Hilbert

La transformée de Hilbert d'une fonction $f(t)$ est définie par la valeur principale de Cauchy (Roddier, 1971):

$$(Hf)(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(s)}{t-s} ds \quad (\text{EN.7})$$

Dans le domaine de Fourier, cette transformée s'exprime comme un simple déphasage de $\pm\pi/4$ (filtre en quadrature):

$$(\hat{H}f)(\omega) = -i \operatorname{sgn}(\omega) \hat{f}(\omega) \quad (\text{EN.8})$$

N.6 - Produit de Kronecker

Etant donné deux matrices $\underline{\mathbf{A}}(m, n)$ et $\underline{\mathbf{B}}(k, l)$, leur produit de Kronecker (Regalia and Mitra, 1989) est une matrice $(m.k,n.l)$:

$$\underline{\mathbf{A}} \otimes \underline{\mathbf{B}} = \begin{bmatrix} \underline{\mathbf{A}}b_{1,1} & \underline{\mathbf{A}}b_{1,2} & \dots & \underline{\mathbf{A}}b_{1,l} \\ \underline{\mathbf{A}}b_{2,1} & \underline{\mathbf{A}}b_{2,2} & \dots & \underline{\mathbf{A}}b_{2,l} \\ \dots & \dots & \dots & \dots \\ \underline{\mathbf{A}}b_{k,1} & \underline{\mathbf{A}}b_{k,2} & \dots & \underline{\mathbf{A}}b_{k,l} \end{bmatrix} \quad (\text{EN.9})$$

Parmi les propriétés les plus évidentes:

$$\begin{aligned} (\underline{\mathbf{A}} \otimes \underline{\mathbf{B}}) \otimes \underline{\mathbf{C}} &= \underline{\mathbf{A}} \otimes (\underline{\mathbf{B}} \otimes \underline{\mathbf{C}}) \\ (\underline{\mathbf{A}} \otimes \underline{\mathbf{B}})(\underline{\mathbf{C}} \otimes \underline{\mathbf{D}}) &= (\underline{\mathbf{A}}\underline{\mathbf{C}}) \otimes (\underline{\mathbf{B}}\underline{\mathbf{D}}) \\ \underline{\mathbf{A}} \otimes \underline{\mathbf{D}} &= (\underline{\mathbf{A}} \otimes \underline{\mathbf{Id}}_l)(\underline{\mathbf{Id}}_n \otimes \underline{\mathbf{D}}) \end{aligned} \quad (\text{EN.10})$$

(Dans les deux dernières relations, les matrices $\underline{\mathbf{C}}$ et $\underline{\mathbf{D}}$ sont de dimensions (n,q) et (l,s))

Bibliographie

- Bardan, V., 1989, Sampling two-dimensional seismic data and their Radon transforms: *Geophysics*, **54**, 1318-1325.
- Bardan, V., 1997, A hexagonal sampling grid for 3D recording and processing of seismic data: *Geophysical Prospecting*, **45**, 819-839.
- Backus, M. M., 1959, Water-reverberations, their nature and elimination: *Geophysics*, **24**, 233-261.
- Beasley, C. J., and Klotz, R., 1992, Equalization of DMO for irregular spatial sampling: 54th Mtg. Eur. Assoc. Expl Geophys., Abstracts, Western Geophys. Co., 232-233.
- Bube, K. P., and Langan, R. T., 1997, Hybrid L1/L2 minimization with applications to tomography: *Geophysics*, **62**, 1183-1195.
- Burg, J., P., 1975, Maximum entropy spectral analysis: Ph. D Thesis, Stanford University.
- Canadas, G., Desaunay, S. and de Bazelaire, E., 1991, Multiple elimination via inversion velocity analysis and shifted hyperbolas: 53rd Mtg. Eur. Assoc. Expl. Geophys., Abstracts, 18-19.
- Canadas, G., and Dunand, J.-P., 1995, AVO study in the parabolic Radon transform domain: 65th Annual Meeting, SEG, Expanded Abstracts, 1370-1372.
- Carron, D., 1989, Solutions parcimonieuses à certains problèmes inverses discrets sous-déterminés en géophysique pétrolière.: Thèse de Doctorat, Université de Paris VI.
- Chapman, C. H., 1981, Generalized Radon transforms and slant stacks: *Geophysical Journal of the Royal Astronomical Society*, **66**, 445-453.
- Claerbout, J. F., 1975, Slant Stacks and Radial Traces, *in* Gardner, H. F. and Lu, L., Eds., *Slant Stack Processing: Soc. Expl. Geophys.*, 73-84
- Claerbout, J. F., 1985, *Imaging the earth's interior: Blackwell Scientific Publications.*
- Claerbout, J., 1992, *Earth Soundings Analysis - Processing Versus Inversion: Blackwell Scientific Publications, Cambridge, USA*
- Claerbout, J., and Nichols, D., 1994, Spectral Preconditioning: SEP report, **82**, 183-186.
- Darce, G., 1989, Iterative L1 deconvolution: SEP report, **61**, 281-302.
- Darce, G., 1990a, Fast parabolic transforms: SEP report, **65**, 99-112.
- Darce, G., 1990b, Spatial Interpolation using a fast parabolic transform: 60th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1647-1650
- de Bazelaire, E., 1988, Normal MoveOut revisited: Inhomogeneous media and curved interfaces: *Geophysics*, **53**, 143-157.
- Dietrich, M., 1988, Modeling of marine seismic profiles in the t - x and τ - p domains: *Geophysics*, **53**, 453-465
- Donati, M. S., Martin, N. W., and Stewart, R. R., 1996, 3D tau-p filtering: EAEG 58th Conference and Technical Exhibition, Extended Abstracts, Session B004.
- Duijndam, A., 1987, Detailed Bayesian Inversion of seismic data: PhD thesis, Delft University of Technology.
- Duijndam, A. J., Schils, J., and Schonewille, M. A., 1995, Reconstruction of bandlimited, irregularly-sampled signals: Soc. Expl. Geophys. 65th Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1350-1353.
- Duijndam, A. J. W., and Schonewille, M. A., 1997, Fourier-Radon reconstruction of sparsely sampled signals in the midpoint-offset domain: EAEG 59th Conference and Technical Exhibition, Extended Abstracts, Session A036.
- Durrani, T. S., and Bisset, D., 1984, The Radon transform and its properties: *Geophysics*, **49**, 1180-1187.
- Durrani, J. A., 1991, Multiple attenuation by predictive deconvolution in the p - τ domain: 53rd Mtg. Eur. Assoc. Expl. Geophys., Abstracts, 26-27

- Feichtinger, H. G., Grochening, K., and Strohmer, T., 1995, Efficient numerical methods in non-uniform sampling theory: *Numerische Mathematik*, **69**, 423-440.
- Fokkema, J. T., Van Den Berg, P. M., and Vissinga, M., 1992, On the computation of Radon transforms of seismic data: *Journal of seismic exploration*, **1**, 93-105.
- Foster, J., and Mosher, C., 1992, Suppression of multiple reflections using the Radon transform: *Geophysics*, **57**, 386-395.
- Gavin, P., Mars, J. and Essebar, A., 1994, Wavefield decomposition on offset VSP data using parametric separation: 56th Mtg. Eur. Assoc. Expl. Geophys., Extended Abstracts, Session:G030.
- Gersztenkorn, A., Bednar, J. B., and Lines, L. R., 1986, Robust iterative inversion for the one-dimensional acoustic wave equation: *Geophysics*, **51**, 357-368.
- Gulunay, N., 1990, F-X domain least-squares tau-p and tau-q: 60th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1607-1610.
- Hampson, D., 1986, Inverse velocity stacking for multiple elimination: *Journal of Can. Soc. Expl. Geophys.*, **22**, 44-55.
- Hampson, D., 1987, The discrete Radon Transform: a new tool for image enhancement and noise Suppression: 57th Annual Internat. Mtg., SEG, Expanded Abstracts,
- Helgason, S., 1980, *The Radon Transform*: Birkhauser, Boston
- Hindrinks, K., Duijndam, A., and Petersen, S., 1997a, Reconstruction of bandlimited signals, irregularly sampled along two spatial directions: EAEG 59th Conference and Technical Exhibition, Extended Abstracts, Session A033.
- Hindrinks, K., Duijndam, A., and Schonewille, M. A., 1997b, Reconstruction of two-dimensionnal irregularly sampled wavefields: 67th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1163-1166.
- Huard, I., and Spitz, S., 1997, An application of 3D filtering to the restoration of missing traces: EAEG 59th Conference and Technical Exhibition, Extended Abstracts, Session:A038.
- Hubral, P., 1991, Slant stack migration, *in* Gardner, G. H. F., and Lu, L., Ed., *Slant stack processing*: Soc. Expl. Geophys., 111-117.
- Hugonnet, P., and Canadas, G., 1995, Aliasing in the Parabolic Radon Transform: 65th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1366-1369.
- Hugonnet, P., and Canadas, G., 1997, Aliasing in the Parabolic Radon Transform: 67th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1111-1114.
- Jeffery, R. W., and Kelamis, P. G., 1994, Cascaded linear and parabolic Radon noise suppression - A technique for 3D seismic data improvement: 56th Mtg. Eur. Assoc. Expl Geophys., Extended Abstracts, Session:H050.
- Kabir, M. M. N., 1992, Implementation and applications of the Generalized Radon Transform: M.Sc. Thesis, Delft University.
- Kabir, M. M. N., and Vershuur, D. J., 1995, Restoration of missing offsets by parabolic Radon transform: *Geophysical Prospecting*, **43**, 347-368.
- Kappus, M. E., Harding, A. J., and Orcutt, J. A., 1990, A comparison of tau-p transform methods: *Geophysics*, **55**, 1202-1215.
- Kinneging, N. A., Duijndam, A. J. W., and Ongkiehong, L., 1991, Protection against spatial aliasing by means of wavefield reconstruction: 61st Annual Meeting, Soc. Expl. Geoph., Expanded Abstracts, 1373-1376.
- Kormendi, F., and Dietrich, M., 1991, Non linear inversion of plane wave seismograms in stratified elastic media, *Geophysics*, **56**, 664-674.

- Kostov, C., 1989, Finite-aperture slant-stack transform: SEP report, **61**, 261-279.
- Kostov, C., 1990, Toeplitz structure in Slant-Stack Inversion: 60th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1618-1621.
- Lokshantov, D. E., 1993, Adaptive multiple suppression in tau-p domain: 63rd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1086-1089
- Lu, L., 1991, Interpolation, *in* Gardner, G.H.F. and Lu, L., Ed., Slant-stack processing: Soc. Expl. Geophys., 302-303
- Lumley, D. E., 1994, Estimating a pseudo-unitary operator for velocity stack inversion: SEP report, **82**, 63-78.
- Lumley, D. E., Nichols, D., and Rekdal, T., 1995, Amplitude-preserved multiple suppression: 65th Internat. Ann. Mtg., Soc. Expl. Geoph., Expanded Abstracts, 1460-1463.
- Maroof, S. I., and Gravely, C. J., 1984, Aliasing and tau-p: 54th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 623-625.
- Marvasti, F., Analoui, M., and Gamshadzahi, M., 1991, Recovery of signals from non-uniform samples using iterative methods: IEEE transactions on signal processing, **39**, 872-878.
- Nichols, D., 1990, Estimation of missing data by least squares: SEP report, **65**, 271-293.
- Nichols, D., 1992, Dealiasing band limited data using a spectral continuity constraint: 62nd Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1183-1186.
- Nichols, D., 1994, Velocity-Stack inversion using Lp norms: SEP report, **82**, 1-16.
- Ottolini, R., and Claerbout, J. F., 1984, The migration of common midpoint slant stacks: Geophysics, **49**, 237-249.
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., 1992, Numerical Recipes in Fortran: the art of scientific computing: Cambridge University Press, Cambridge.
- Radon, J., 1917, Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten: Ber. Verh. Sächs. Akad. Wiss, Leipzig, Math-Nat. **69**, 262-277.
- Regalia, P. A., and Mitra, S. K., 1989, Kronecker Products, Unitary Matrices and signal processing applications: SIAM Review, **31**, 586-613.
- Riabinkin, L. A., 1957, Fundamentals of resolving power of controlled directional reception (CDR) of seismic waves, *in* Gardner, G. H. F. and Lu, L., Eds., Slant Stack Processing: Soc. Expl. Geophys., 390-418
- Rieber, F., 1936, A new reflection system with controlled directional sensitivity: Geophysics, **1**, 97-106.
- Rieber, F., 1937, Visual presentation of elastic wave patterns under various structural conditions: Geophysics, **1**, 196-218.
- Robinson, E. A., and Treitel, S., 1980, Geophysical Signal Analysis: Prentice Hall, Englewood Cliffs, NJ
- Roddier, 1971, Distributions et Transformées de Fourier: McGraw-Hill, Paris
- Ronen, J., 1987, Wave equation trace interpolation: Geophysics, **52**, 973-984.
- Ronen, S., Nichols, D., Bale, R.A. and Ferber, R., 1995, Dealiasing DMO: Good-pass, bad pass, and unconstrained: 65th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 743-746.
- Sacchi, M. D., and Ulrych, T. J., 1994, High resolution velocity gather: 64th Annual Meeting, Soc. Expl. Geophys., Expanded Abstracts, 1469-1472.
- Sacchi, M. D., and Ulrych, T. J., 1995, Model re-weighted least-squares Radon operators: 65th Internat. Annual Mtg, Soc. Expl. Geophys., Expanded Abstracts, 616-618.
- Scales, J. A., and Gersztenkorn, A., 1987, Robust methods in inverse theory, *in* Lines, L. R. and Scales, J. A., Eds., Geophysical imaging, symposium of Geophysical Society of Tulsa: Soc. Expl. Geophys., Tulsa, 25-50

- Shin, Y., and Sumner, B. L., 1994, Least-squares dip filtering and regridding of 3D seismic data: 56th Mtg. Eur. Assoc. Expl Geophys., Extended Abstracts, Session:H053.
- Singh, S. C., West, G. F., and Chapman, C. H., 1989, On plane-wave decomposition : Alias removal: *Geophysics*, **54**, 1339-1343.
- Soubaras, R., 1997, The necessary and sufficient condition for lossless sampling: EAEG 59th Conference and Technical Exhibition, Extended Abstracts, Session:A037.
- Spitz, S., 1989, Trace interpolation beyond aliasing and without picking: 59th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1121.
- Spitz, S., 1990, 3-D seismic interpolation in the f-xy domain: 60th Annual Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, 1641-1643
- Spitz, S., 1991, Seismic trace interpolation in the F-X domain: *Geophysics*, **56**, 785-794.
- Stoffa, P. L., Buhl, P., Diebold, J. B., and Wenzel, F., 1981, Direct mapping of seismic data to the domain of intercept and ray parameter - A plane-wave decomposition: *Geophysics*, **46**, 255-267.
- Tarantola, A., 1987, *Inverse Problem Theory*: Elsevier Science Publisher, Amsterdam
- Thorson, J. R., 1984, *Velocity and Slant Stack inversion*: Ph.D Thesis, Stanford University.
- Thorson, J. R., and Claerbout, J. F., 1985, Velocity-stack and slant-stack stochastic inversion: *Geophysics*, **50**, 2727-2741.
- Treitel, S., Gutowski, P. R., and Wagner, D. E., 1982, Plane-wave decomposition of seismograms: *Geophysics*, **47**, 1375-1401.
- Trorey, A. W., 1961, The information content of a Rieber sonogram: *Geophysics*, **26**, 761-764.
- Turner, G., 1990, Aliasing in the tau-p transform and the removal of spatially aliased coherent noise: *Geophysics*, **55**, 1496-1503.
- Wiggins, J. W., and Abma, R., 1990a, Using the Parabolic Radon Transform as a moveout filter: 52nd EAEG Annual Meeting, Abstracts, 69-70.
- Wiggins, J. W., and Abma, R., 1990b, Using the Parabolic Radon Transform as a moveout filter: report, Western Geophysical, Houston TX.
- Wiggins, R., 1985, Entropy guided deconvolution: *Geophysics*, **50**, 2720-2726.
- Yilmaz, O., and Taner, M. T., 1994, Discrete plane-wave decomposition by least-mean-square-error method: *Geophysics*, **59**, 973-982.
- Zhou, B., and Greenhalgh, A., 1994, Linear and parabolic tau-p transforms revisited: *Geophysics*, **59**, 1133-1149.
- Zhou, B., and Greenhalgh, A., 1996, Multiple suppression by 2D filtering in the parabolic tau-p domain: a wave equation-based method, *Geophysical Prospecting*, **44**, 375-401.