Eulerian approach of Hamilton-Jacobi equation with a discontinuous Galerkin method in heterogeneous anisotropic medium : Application to seismic imaging

Philippe Le Bouteiller

Communauté UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTE UNIVERSITE GRENOBLE ALPES

Spécialité : **Sciences de la Terre et de l'Univers et de l'Environnement (CESTUE)**

Arrêté ministériel : 25 mai 2016

Présentée par

Philippe LE BOUTEILLER

Thèse dirigée par **Jean VIRIEUX, Professeur Émérite, UGA**, et codirigée par **Ludovic MÉTIVIER, Chargé de Recherche, CNRS**

préparée au sein du laboratoire **Institut des Sciences de la Terre** dans **l'École Doctorale Terre, Univers, Environnement**

Approche eulérienne de l'équation de Hamilton-Jacobi par une méthode Galerkine discontinue en milieu hétérogène anisotrope : Application à l'imagerie sismique

Eulerian approach of Hamilton-Jacobi equation with a discontinuous Galerkin method in heterogeneous anisotropic medium: Application to seismic imaging

Thèse soutenue publiquement le **6 décembre 2018**, devant le jury composé de :

Monsieur Emmanuel MAITRE
Professeur, Université Grenoble Alpes, Président
Monsieur Dirk GAJEWSKI
Professeur, Université de Hambourg (Allemagne), Rapporteur
Monsieur Jean-Marie MIREBEAU
Chargé de Recherche, CNRS Délégation Ile-de-France Sud, Rapporteur
Monsieur Hervé CHAURIS
Professeur, MINES ParisTech, Examinateur
Monsieur Jean VIRIEUX
Professeur Émérite, Université Grenoble Alpes, Directeur de thèse
Monsieur Ludovic MÉTIVIER
Chargé de Recherche, CNRS Délégation Alpes, Co-directeur de thèse

SIGILLUM UNIVERSITATIS GRATIANOPOLITANAE
VERI TAS
LIBE RABIT
1339

À Tchouquette

Remerciements

Mon intérêt pour la géophysique remonte à mon passage à l'école des Mines, en 2010, au sein de l'option Géosciences, alors encadrée par Pascal Podvin. Malgré un stage à Fontainebleau au cours duquel j'ai fait la rencontre d'Alexandrine, Mark et Hervé, j'avais ensuite décidé de suivre une autre route. Il aura finalement fallu 4 ans pour que l'idée de faire une thèse remonte à la surface. Cher Hervé, je te remercie pour nos échanges réguliers, jusqu'à ce jour où tu m'as mis en contact et recommandé auprès de Jean. C'est grâce à toi que je suis parti pour Grenoble, emportant toute la famille dans mes bagages. Je fais un clin d'oeil au passage à Philippe M., mentor et conseiller carrière personnel !

Sur le plan scientifique, ma thèse s'est passée dans les meilleures conditions possible. Je remercie ici mes encadrants, Jean Virieux et Ludovic Métivier. Jean, ton exigence scientifique, l'étendue de ton expérience et la richesse de nos échanges m'ont aidé à tracer une voie pas évidente au milieu d'un champ de recherche déjà largement exploré et pourtant jamais conquis. Ludovic, ta probité intellectuelle, ta rigueur de mathématicien et tes corrections de mes papiers en anglais maladroit m'auront été d'un grand secours. Je remercie également le consortium SEISCOPE, mes encadrants et Romain Brossier en tête, pour le soutien inconditionnel à mes recherches, alors même que je n'ai jamais prononcé le mot « waveform ». Merci à l'équipe ondes et ses pizzas-séminaires, merci enfin au laboratoire ISTerre et à tous ceux qui œuvrent pour que les travaux de recherche y soient menés dans de bonnes conditions.

Sur le plan humain, un laboratoire de recherche rassemble et mélange des profils et des personnalités d'une grande variété, donnant l'occasion de rencontres d'une grande richesse. Je pense aux nombreux échanges à la cafétéria, à table, en faisant la queue pour les micro-ondes ou encore d'un évier à l'autre pendant la vaisselle. Je pense également à toutes les amitiés nouées entre les doctorants. Merci à mes co-bureaux successifs pour les bons moments passés : Johann, ses Pépitos et ses clics frénétiques, Blandine et sa facilité de rédaction déconcertante, Capucine, son accent chantant, ses déboires avec EDF et ses idoles masculines, Chloé, ses T-shirts collectors et son art du rangement… Merci à Jérémie, sommet inatteignable de mon horizon scientifique, aux publications pléthoriques et à la capacité de travail impressionnante. J'espère de tout cœur que tu pourras trouver équilibre de vie et sérénité dans un environnement intellectuel stimulant.

En arrivant à Grenoble nous nous sommes vite sentis chez nous. Clin d'œil à Joseph pour le logement. Merci à nos voisins, à Philippe G. en particulier. Merci à Caro et Gio pour l'atterrissage en douceur. Et parmi d'autres coïncidences, quelle joie de me retrouver dans le même labo que David !

Vivre à Grenoble sans profiter de la montagne, cela serait dommage. Merci à Marc pour l'initiation au ski de fond. Un jour je pourrai peut-être doubler Capucine... Il se murmure que l'UT4M avec Antoine serait pour 2020. Quant à l'Ultimate, il serait temps de relancer les Flying Earth pour garder la forme ! Camille, Laurent, Pascual, au boulot !

L'aventure MT180 a été très marquante pour moi, et je remercie ici tous ceux qui m'ont encouragé, accompagné et suivi au fil des mois. À tous ceux qui ont subi les nombreuses répétitions – collègues,

amis, inconnus – merci pour les nombreux retours qui m'ont permis d'améliorer toujours plus ma prestation. Merci à Ludovic, mon coach, pour son approche qui propose à chacun d'oser se révéler tel qu'il est face au public.

Une année 2018 réussie : les médailles aux JO de Martin Fourcade, la Coupe du Monde, MT180, MT tout court… que reste-t-il pour 2019 ? Plus excitant que le Brexit ou que la menace de crise financière mondiale, deux belles aventures commencent : Tchouquette, que nous avons hâte de découvrir, et HYMAG'IN ! Merci à Camille et Céline de m'avoir proposé de les rejoindre dans l'aventure. Un nouveau défi s'offre à nous. Merci à toute l'équipe pour son accueil chaleureux. Et merci à nouveau à Philippe M. pour la validation du choix !

Enfin, un grand merci à toute ma famille. A Maman, qui nous soutient toujours dans nos choix. A Papa, pour qui je crois que ça a été un peu plus dur. Merci à Gabriel et Laetitia, qui grandissent vite et nous font redécouvrir les mystères de la vie.

Un immense merci, Marguerite, pour ton soutien de tous les jours.

Grenoble, le 22 janvier 2019

Résumé

Pouvoir déterminer la structure et la composition de l'intérieur de la Terre est un enjeu scientifique fondamental, pour la compréhension de l'organisation de la Terre profonde, des mécanismes des séismes et leur localisation en lien avec la prévention du risque sismique, pour la détection et l'exploitation des ressources naturelles telles que l'eau ou les hydrocarbures, ou encore pour toutes les activités de construction et de prévention associées au génie civil. Pour cela, les ondes sismiques sont un outil de choix. L'utilisation d'une approximation haute fréquence pour la modélisation de la propagation des ondes est avantageuse en termes de coût de calcul dès lors que plusieurs centaines, voire milliers, ou plus de longueurs d'ondes doivent être propagées. À la place de l'équation des ondes linéaire, l'approximation haute fréquence fournit trois équations aux dérivées partielles fondamentales. L'équation Eikonal, non linéaire, permet d'obtenir le temps de trajet. Une deuxième équation fournit l'angle d'émergence. L'équation Eikonal et l'équation des angles appartiennent toutes deux à la grande famille des équations de Hamilton–Jacobi. Enfin, l'équation de transport permet de calculer l'amplitude.

Le tracé des rais sismiques est une technique lagrangienne qui utilise la méthode des caractéristiques pour obtenir un ensemble d'équations différentielles ordinaires à partir de ces équations aux dérivées partielles. Ces équations peuvent être intégrées facilement, donnant ainsi accès au temps de trajet et à l'amplitude le long des rais. Très largement utilisés dans la communauté géophysique du fait de leur simplicité, les outils de tracé de rais ne sont pas pour autant les plus efficaces et les plus robustes en pratique pour des applications d'imagerie et d'inversion haute résolution. En lieu et place, il peut être utile de résoudre directement les équations aux dérivées partielles par une méthode eulérienne. Durant les trois dernières décennies, une multitude de solveurs ont été développés pour l'équation Eikonal, la plupart utilisant la méthode des différences finies. Ces différents travaux visent à obtenir le meilleur compromis entre précision, coût de calcul, robustesse, facilité d'implémentation et souplesse d'utilisation.

Dans cette thèse, je développe une approche différente, se basant principalement sur une méthode Galerkine discontinue. Dans le champ des mathématiques, cette méthode a été largement utilisée pour résoudre les lois de conservation et les équations de Hamilton–Jacobi. Très peu de travaux ont porté sur l'utilisation de cette méthode pour la résolution de l'équation Eikonal statique dans un contexte géophysique, et ce malgré le haut niveau de précision qu'elle apporte. C'est pourquoi, en me basant sur des travaux mathématiques, je propose un nouveau solveur Eikonal adapté au contexte géophysique. Les milieux hétérogènes complexes, anisotropes, et incluant des variations topographiques sont correctement pris en compte, avec une précision sans précédent. En y intégrant de manière robuste une stratégie de balayage rapide, je montre que ce solveur présente une très grande efficacité en deux comme en trois dimensions.

J'utilise également ce solveur pour calculer l'angle d'émergence. Je développe par ailleurs un solveur voisin en volumes finis pour la résolution de l'équation de transport, permettant ainsi le calcul

de l'amplitude. La variable d'état adjoint pour la tomographie sismique des temps et des pentes vérifiant une équation de transport semblable, je montre qu'on peut également la calculer à l'aide de ce solveur en volumes finis. En conséquence, je propose et analyse un ensemble consistant de solveurs pour la communauté géophysique. Ces outils devraient s'avérer utiles pour une large palette d'applications. Finalement, en guise d'illustration, je les utilise dans des schémas d'imagerie sismique, dans le but de démontrer le bénéfice apporté par une approximation haute fréquence dans ce type de schémas.

Abstract

Recovering information on the structure and the composition of the Earth's interior is a fundamental issue for a large range of applications, from planetology to seismology, natural resources assessment, and civil engineering. Seismic waves are a very powerful tool for that purpose. Using a high-frequency approximation for the numerical modeling of seismic wave propagation is computationally advantageous when hundreds, thousands, or more of wavelengths have to be propagated. Instead of the linear wave equation, the high-frequency approximation yields three fundamental partial differential equations. The nonlinear Eikonal equation leads to traveltime. A second equation is derived for the take-off angle. Both Eikonal and angle equations belong to the wide Hamilton–Jacobi family of equations. In addition, the transport equation leads to the amplitude.

As a Lagrangian approach, seismic ray tracing employs the method of characteristics to derive a set of ordinary differential equations from these partial differential equations. They can be easily integrated, thus yielding traveltime and amplitude along rays. Widely used in the geophysical community for their simplicity, the ray-tracing tools might not be the most efficient and robust ones for practical high-resolution imaging and inversion applications. Instead, it might be desirable to directly solve the partial differential equations in an Eulerian way. In the three last decades, plenty of Eikonal solvers have been designed, mostly based on finite-difference methods. Successive works try to find the best compromise between accuracy, computational efficiency, robustness, ease of implementation, and versatility.

In this thesis, I develop a different approach, mainly based on the discontinuous Galerkin method. This method has been intensively used in the mathematical field for solving conservation laws and time-dependent Hamilton–Jacobi equations. Only few investigations have been done regarding its use for solving the static Eikonal equation in a geophysical context, despite the high level of accuracy allowed by this method. Therefore, improving upon mathematical studies, I propose a new Eikonal solver suitable for the geophysical context. Complex heterogeneous anisotropic media with non-flat topographies are correctly handled, with an unprecedented accuracy. Combined with a fast-sweeping strategy in a robust way, I show that this new solver exhibits a high computational efficiency, in two dimensions as well as in three dimensions.

I also employ this solver for the computation of the take-off angle. I design an additional finite-volume solver for solving the transport equation, leading to the computation of amplitude. With this solver, I also consider the computation of the adjoint-state variable for seismic tomography, since it satisfies a similar transport equation. Eventually, I propose a whole set of consistent solvers to the geophysical community. These tools should be useful in a wide range of applications. As an illustration, I finally use them in advanced seismic imaging schemes, in order to demonstrate the benefit brought by the high-frequency approximation in this kind of schemes.

Contents

General introduction

Imaging the Earth's interior

Several scales for several goals

Knowing the structure and composition of the interior of the Earth is essential for a large amount of concerns, ranging from planetology to natural resources assessment and civil engineering. At very large scales, the study of its layered structure and the inferred composition of each layer from the crust to the inner core yields crucial information regarding the formation of the Earth 4.5 billion years ago. Understanding the internal dynamics of the different layers provides an explanation for various phenomena such as the Earth's magnetic field due to convection within the liquid metallic outer core, as well as plate tectonics governed by mantle convection, which is at the origin of orogeny, subduction zones, and the majority of earthquakes and volcanoes. At regional scales, Earth scientists from various disciplinary fields try to better understand these phenomena. For instance, they study the characteristics and the propagation modalities of earthquakes. They try to estimate the size of magma chambers of volcanoes and to describe the characteristics of their eruptions. They also assess the associated risks for the human populations. At smaller scales, several industrial activities rely upon the knowledge of the structure of the underground. Retrieving information coming from dozens to thousands meters depth is the essence of exploration geophysics, aimed at finding natural resources such as oil and gas and extract them. The purpose of geotechnical studies is to acquire knowledge on the first meters of the Earth's crust. These studies involve various fields such as rock mechanics, geology, hydrology, and civil engineering. Their aim can be, for instance, to investigate on the stability of dam structures, or to design suitable foundations for buildings. This knowledge is also fundamental for environmental sustainability when it comes to groundwater pollution monitoring, ultimate waste management, and CO_2 sequestration.

How to look at the Earth's deep interior?

Since the Earth is not translucent for light, we cannot examine it with our eyes. Sparse direct observations can be obtained by drilling, although costly and limited in depth. The Kola Superdeep Borehole was drilled in 1989 on the Kola Peninsula in the Soviet Union. It reached "only" 12.262 km depth, which represents 0.2% of the Earth radius (6400 km), and this is the deepest borehole ever drilled so far. The drilling could not go deeper because of the high temperature encountered at this depth, 180°C, that would make the driller melt. Therefore, we need other tools to investigate the interior of the Earth. At the beginning of the twentieth century, physicists began to study the Earth's interior by the use of earthquakes. When earthquakes of high magnitude occur, seismic waves propagate inside the Earth, and can be recorded at the surface after their travel. Compressional (P-) waves travel across liquids and

solids, while shear (S-) waves only propagate into solids. Using recordings from all over the globe, Oldham built a first seismological model of the Earth in 1906, highlighting a first large discontinuity in the deep interior (Oldham, 1906). Mohorovičić observed a shallow discontinuity (few tens of kilometers) in 1909 (Mohorovičić, 1909), which was later understood as the interface between the crust and the mantle. In 1913, Gutenberg brought to light the existence of a core from the observation of shadow zones, at a certain angle at the surface from the source of an earthquake, where P waves are not observed (Gutenberg, 1914). One can also mention, among others, Inge Lehmann, who divided in 1936 the core into an inner and an outer part (Lehmann, 1936). Multiple internal reflections and conversions between P- and S- waves have been studied and classified by seismologists who are able to recognize them on recordings (see figure 1). We refer the reader to the comprehensive review in Dziewonski and Romanowicz (2015) for more historical details.

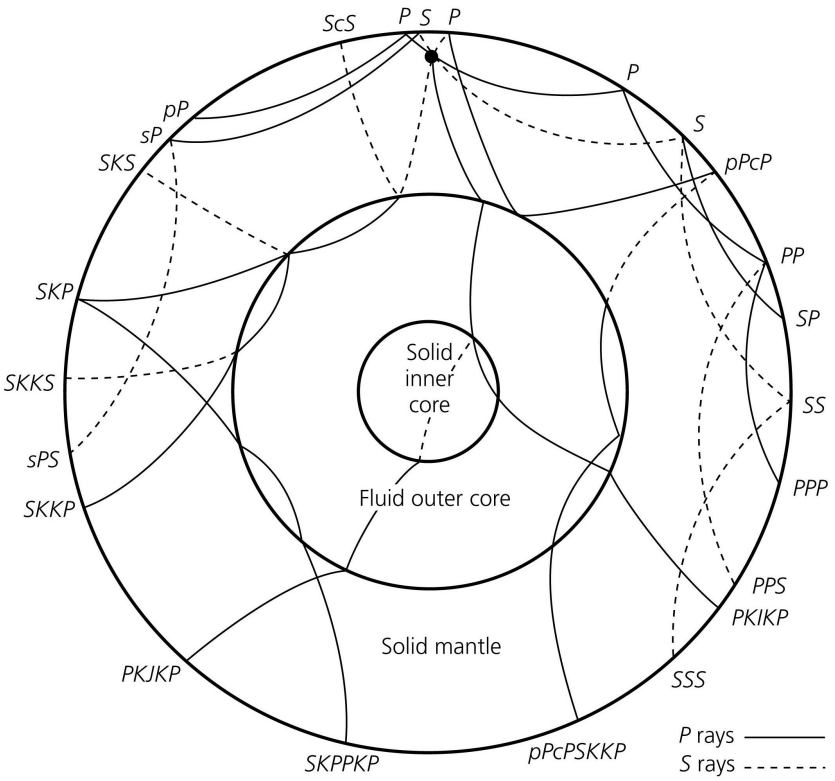


Figure 1: Simplified internal structure of the Earth and examples of seismic phases. Their names are given with respect to the phase types (P and S) and the number of reflections occurring at the discontinuities: Inner/outer core, core/mantle, surface (from Stein and Wysession, 2003).

Thus, since the beginning of the twentieth century, seismic waves, as a natural phenomenon triggered by earthquakes, have been used as a tool for inferring the Earth's interior structure. This is enabled by the fact that these waves strongly interact with rocks within the Earth. When they reach the surface, a large amount of information related to the medium they have traveled in is encoded in the recorded signal. In seismology, the above-mentioned early descriptions of the Earth's interior were based on wave traveltimes, which means that only a part of the signal was interpreted. Later, finer descriptions were provided thanks to the use of the whole signal. For instance, lateral variations of the Earth structure inside the upper mantle were described by Woodhouse and Dziewonski (1984), using a method based on normal modes summation, able to interpret the whole seismic waveform.

Besides seismic waves, other physical phenomena are used in geophysics for the investigation of the Earth's interior. For instance, borehole, land, airborne or satellite measurements of gravity can be used to investigate rock density distribution in the crust and the upper mantle (LaFehr, 1983). Measurements of the geomagnetic field and its secular variation (changes on time scales of about a year) gives information on fluid motion in the Earth's core (Gubbins, 1996). Surface nuclear magnetic resonance (SNMR) measurements are used to estimate aquifer properties such as porosity and hydraulic conductivity (Legchenko, 2013). Detection of the neutrinos emitted in the Earth in decay of radionuclide (geoneutrinos) yields geological information related to the abundance and spatial distribution of radionuclides in the Earth (Araki et al., 2005). However, the seismic methods are the only ones able to investigate at all scales, which makes them a widespread tool in geophysics.

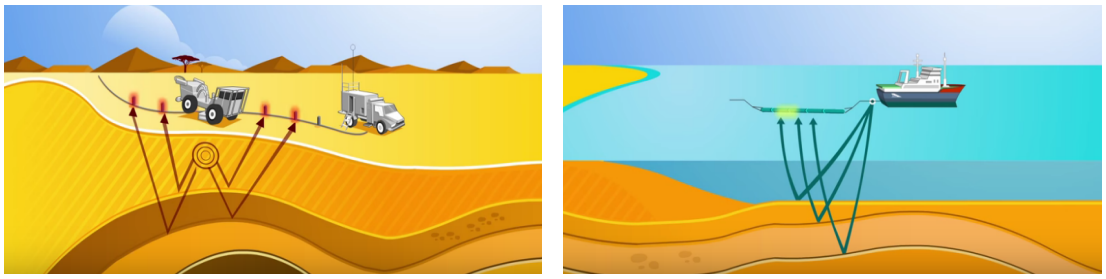
Finding ways to interpret the seismic signal in order to retrieve information on the Earth structure is the core activity of global and regional seismology, which is mostly performed in the academy. It is also the core activity of seismic exploration, which refers to the use of seismic waves for natural resources assessment purposes. An overview of this industrial field is given next.

Seismic exploration

Seismic methods are intensively used for hydrocarbon (oil and gas) exploration. They can also be operated in the exploration of other natural resources like minerals or nuclear fuel. Developed since the 1930's, it is still a very active field of research, with challenges associated with the increasing level of resolution needed. Seismology generally uses waves generated by natural earthquakes, and, more recently, seismic noise coming mainly from oceanic waves and storms, as tools for imaging the Earth's interior. On the opposite, a typical seismic survey utilizes so-called controlled sources, such as dynamite blasts or seismic vibrators for land acquisition, or air guns for marine acquisition. These sources are triggered at controlled times and positions, while an array of seismic recorders (receivers) is deployed over the region of interest. Seismic waves propagate down into the ground and interact with the interfaces between sedimentary layers. A part of the energy travels back to the receivers. Depending on the main interaction type that is tracked, the technique is called seismic reflection or seismic refraction. In seismic reflection, receivers are deployed at the surface and record the upgoing waves that are reflected at the interfaces. A land acquisition may involve truck-mounted seismic vibrators and arrays of fixed receivers (see figure 2a). Meanwhile, a typical marine survey involves one or several vessels towing the seismic source as well as multiple hydrophones capable to measure the pressure field. They are mounted on streamer cables with lengths ranging from 3000 to 6000 meters (see figure 2b). Marine acquisition might also involve seismic receivers fixed at the seafloor, recording the ground displacement, and mounted on ocean bottom cables.

High resolution, big data, and computational efficiency

In seismology as well as in seismic exploration, a general trend is the acquisition of more and more data. The first expected benefit is an increase of the resolution, which means a decrease of the size of the smaller structures that might be retrieved from the data. In addition, this should yield a reduction of the uncertainty associated with the estimation. In this context, three big challenges must be faced. The first one is related to the need of high-quality and high-density data acquisition. The second is related to big data management, and the large computational infrastructure required to treat all the data. The third one is the development of high-performance processing algorithms. Data acquisition involves, for instance, designing, producing, and using high-sensitivity receivers, as well as designing and deploying



(a) Onshore (b) Offshore

Figure 2: Schematic of an onshore and offshore setup for a seismic survey (source: Total).

dense seismic acquisition systems with a large number of sources and receivers. A typical seismic survey may involve few hundreds of thousands of traces (seismograms) per square kilometer. Hence, the second challenge is to be able to store and process a considerable amount of data. It is worth mentioning that Earth imaging has a significant imprint in the world's biggest supercomputer lists. Major oil and gas companies possess their own supercomputers which generally rank among the first private supercomputers in the TOP500 list, like Total's Pangea, which ranked #11 in the June 2013 list with an initial capacity of 2.1 petaflops (2.1 quadrillion floating-point operations per second), and now ranking #30 in the June 2018 list with a capacity of 5.3 petaflops (`www.top500.org`). Finally, data processing needs not only big infrastructure, but also efficient numerical algorithms. High-resolution imaging strategies make the most of the data and the available infrastructure in order to obtain the best images with a running time as short as possible.

Basics of inversion: forward and inverse problems

Using seismic waves to study the Earth's interior requires to understand how seismic waves propagate and interact with the medium mechanical properties. Several physical models have been proposed to describe wave propagation. These models are always based on assumptions and approximations, and they describe the medium by means of different parameters. For instance, the acoustic wave propagation model links the time and space variations of the pressure field to the P-wave velocity and the density. It only describes compressional waves. Meanwhile, the elastic isotropic wave propagation model describes both shear and compressional waves by linking the displacement wavefield to three parameters, namely the density and the Lamé parameters. This might yield more fidelity for representing wave propagation within solids, but also more complexity to design algorithms and increased computational cost. Moreover, a model is designed to be appropriate at a certain scale: for instance, the physical models used for seismic wave propagation from meters to thousands of kilometers are not suitable at atomic scales. Based on a theoretical wave propagation model, one then has to derive a method to simulate this propagation, which is solving the mathematical equations provided by the model in a given medium. For large realistic problems, it is of course not possible to solve these equations analytically. Therefore, numerical methods must be employed, which involve numerical discretization in time and space, and computation of the solution (Carcione et al., 2002). In this thesis, I have been intensively using the discontinuous Galerkin method, which will be described later (Cockburn et al., 2000; Hesthaven and Warburton, 2008). Generally, the accuracy of the result and the computational cost of a numerical method directly depend on the discretization: the finer the better, but also the more expensive. This simulation step is generally referred to as the modeling step, or the forward problem.

This forward operation g amounts to compute some synthetic data $\boldsymbol{d}$ from a given model $\boldsymbol{m}$, formally

$$\boldsymbol{d} = g(\boldsymbol{m}). \tag{1}$$

A forward algorithm must be provided with model properties (wave speed, density, Lamé parameters, for instance) gathered inside $\boldsymbol{m}$. In practice, these properties are initially not known, and the so-called inversion process consists of retrieving those properties from field observations (Tarantola, 1984, 2005). This is performed through an operation

$$\boldsymbol{m} = g^{-1}(\boldsymbol{d}). \tag{2}$$

The operator g might be non-linear, and in practice it can be impossible to compute its exact inverse, due to the ill-posedness of the problem (many models $\boldsymbol{m}$ may yield the same data $\boldsymbol{d}$) and/or the size of the model. Many schemes can be designed in order to approximate the inverse operation. A standard inversion process is the following: starting from an initial guess for medium properties, a first simulation (forward problem) is performed. From this simulation, synthetic observations are obtained: at the locations of the receivers, we extract the results of the simulation. Then, a comparison is performed to measure the discrepancies between the synthetic observations and the field observations. This measure is called misfit function, and writes in a general form

$$\text{res} = ||\boldsymbol{d} - g(\boldsymbol{m})||. \tag{3}$$

Then, the inversion process aims at modifying the initial guess for medium properties in a way that will decrease this residual. The new model would thus better explain the field data. A variety of mathematical methods exist in order to minimize the residual. At the end of the process, which can be performed either all at once or repeated for several iterations, one should be able to retrieve model properties that best explain the data collected in the field. In principle, the more data collected, the more close to the real true medium the retrieved model properties should be.

Context of the thesis

Nature of the observations

Standard seismic recordings, called seismograms or traces, consist of the recording of a signal along a given time window. A typical seismogram from an earthquake beneath Peru is shown in figure 3. From these recordings, the observations used in the inversion process might be of various types. The whole recording can be used: this is called a full waveform approach. In this case, the inversion process aims at fitting time signals from the simulations with the observations (Virieux and Operto, 2009; Guitton and Alkhalifah, 2013; Virieux et al., 2017). Other approaches rely on extracting observables from the records, such as traveltimes (for non-dispersive propagation) or phases (for dispersive propagation). For non-dispersive propagation, the extraction step consists of picking the arrival times, corresponding to the times at which waves reach the receivers. This can be done manually or in an automated way. This new set of observations might be employed within an inversion process, which thus requires a different simulation engine. Instead of solving equations for a pressure field or a displacement field in time and space, one needs to use a different model with associated equations governing the traveltime. This model exists and can be derived following the high-frequency approximation. In this framework, a fundamental equation, called the Eikonal equation, describes the behavior of wave traveltime. This equation belongs to the large family of Hamilton–Jacobi equations. In my work, I consider novel ways to solve these equations. In the geophysical community, robust traveltime inversion tools have been

widely developed and used (Aki and Lee, 1976; Aki et al., 1977; Rawlinson et al., 2010). Compared to full-waveform approaches, these tools keep a reasonable amount of observations to invert for, so that the computational cost is generally moderate. Since they do not make the most of the whole available data, but only the traveltimes extracted from it, these tools achieve a lower resolution than the full waveform approaches. However, the latter need a good initial guess to perform well, so that the former are still widely used in real applications in order to provide such an initial guess.

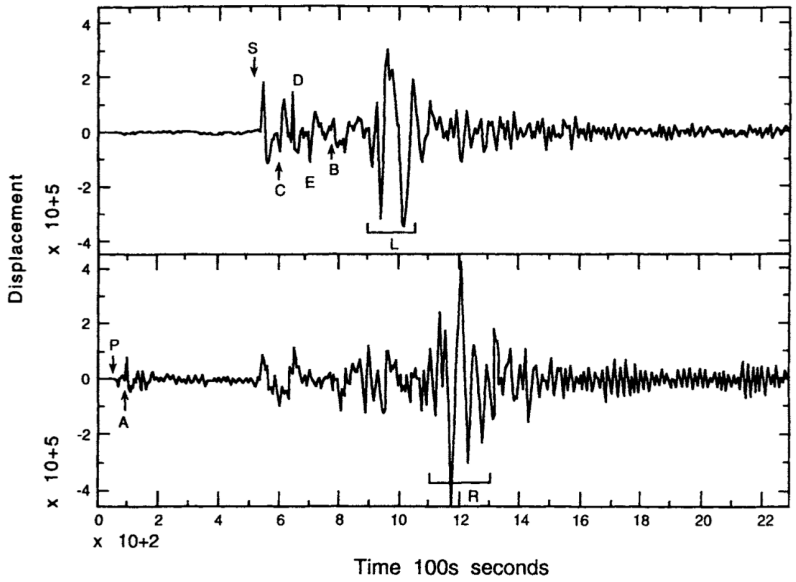


Figure 3: Typical seismogram from a deep earthquake beneath Peru in 1991, recorded in Harvard, Masachusetts: tangential component (top), and vertical component (bottom). Several phases are identified, such as P compressional waves, S shear waves, L Love surface waves, R Rayleigh surface waves, as well as reflected phases. From Lay and Wallace (1995).

Moreover, the high-frequency approximation yields several equations for computing not only traveltime, but also other asymptotic quantities such as amplitude, which describes how the source energy is spread during the propagation in space, as well as take-off angle, which links receiver locations with the emission angle at the source. Like traveltime, these quantities might come into play in several imaging techniques.

Complexity

The Earth's interior is a complex medium, with important heterogeneities and discontinuities at small and large scales. Today, among the most challenging structures that geophysicists try to image, we could cite mantle plumes (Nolet et al., 2007). These large-scale upwellings of abnormally hot material are alleged to occur at the core-mantle boundary and to rise to the crust. We could also mention salt structures at exploration scales (hundreds to thousands of meters) which drastically complicate the imaging of the deeper structures (Peng et al., 2018). The high velocity contrast between water/sediments and salt structure makes most of the seismic energy to be reflected back to the receivers without propagating beneath and below the salt structure. The latter can thus be seen as a screen preventing the imaging of deeper structures. Small-scale heterogeneities of the medium, with typical sizes that are smaller than the wavelength of the propagated signal, do not affect the wave propagation in the same way as large

structures. They result in a mean effect, which can be modeled by the use of an effective, upper-scale medium, which contains anisotropy. This additional complexity expresses through a seismic velocity which depends on the direction of propagation. Embedding anisotropy in a physical model yields more complex equations to solve. Finally, some geometric considerations such as a good handling of the complex topography of the surface of the Earth in certain regions yields specific numerical difficulties.

The higher resolution is pursued in geophysical imaging, the more complexity should be introduced in the models, so that new numerical algorithms are required to take this complexity into account. To give an example, the equations that rule the wave propagation are more complex in an anisotropic medium than in an isotropic one. This is true for both full-waveform and high-frequency formulations.

Objectives

The objectives of this work consist of two main parts. First, it aims at developing efficient and accurate numerical schemes to solve the asymptotic equations for traveltime and derived quantities (amplitude, take-off angle, and adjoint state variable for traveltime tomography) in complex media:

- with complex heterogeneities,
- with anisotropy,
- with complex geometries (topography),
- in 2D as well as in 3D.

Second, I present some applications in the field of seismic imaging in order to illustrate the potential benefit brought by the numerical strategies I develop.

Outline of the manuscript

In this thesis, I present the numerical methods that I develop for seismic modeling and imaging.

- In chapter 1, I introduce the high-frequency approximation, the related equations, and the various numerical methods that have been developed so far in order to solve them, together with their limitations. I show that the need of accuracy and efficiency in complex media makes it desirable to consider alternative techniques to design new numerical solvers.
- In chapter 2, I describe a numerical scheme proposed in the applied mathematics community to solve Hamilton–Jacobi equations for the traveltime using a discontinuous Galerkin discretization. I explain why such a tool is important regarding my objectives, and how I adapt and develop it to make it suitable to traveltime computation in a geophysical context. Finally, I exhibit numerical results that highlight the high level of accuracy reached by this method, together with the high computational time it requires.
- In chapter 3, I describe a numerical tool, namely the fast-sweeping method, which accelerates drastically the computation. I show that I can adapt this tool to the numerical solver I have presented in the previous chapter. I underline with a systematic numerical study that the resulting strategy is more efficient than other existing methods. Thus, it becomes possible to compute traveltime in 3 dimensions, which I illustrate with some numerical results.

- In chapter 4, I describe how I derive accurate numerical solvers for amplitude, take-off angle, and adjoint-state variable, which directly flow from the traveltime solver. All together, these solvers form a new consistent, accurate, and efficient set of tools that should be useful in many geophysical applications.
- In chapter 5, I present the applications that I developed in the field of seismic imaging, in collaboration with the Geophysics team of MINES ParisTech at Fontainebleau, France, and in particular with Hervé Chauris. These applications make the most of the numerical solvers presented in the previous chapters.
- After concluding on my contribution, I give perspectives regarding numerical improvements on the solvers, their potential use in other applications such as traveltime, slope, and diffraction tomography, and the extension to multiple arrivals.

Chapter 1

High-frequency approximation in complex media

Contents

Introduction

In this chapter, I present the theory and the state of the art related to the computation of seismic traveltime and derived quantities like amplitude and take-off angle in the high-frequency approximation framework.

The outline of this chapter is the following:

- In section 1.1, I give the fundamental equations of the wave propagation in elastic and acoustic media, and I describe a high-frequency approximation of this propagation and the resulting equations, in isotropic media.
- In section 1.2, I review three main categories of numerical approaches that have been designed to solve these fundamental equations, respectively the Lagrangian, semi-Lagrangian, and Eulerian

approaches. More details on the Eulerian approach are provided, since it is the method I adopt in the following of this thesis.

- In section 1.3, I extend the theory to anisotropic media, and I review how the numerical approaches are improved in order to tackle anisotropy.

1.1 High-frequency wave modeling: Theory

1.1.1 Wave equations

In this section I will give a brief overview of the derivation of the wave equations in a Cartesian coordinate system. For more details on this derivation and wave propagation, I refer the reader to books such as Fung (1977), Slawinski (2003), and Chapman (2004).

In many geophysical applications, for scales ranging from meters to thousands of kilometers, we need to describe the medium without going into the detail of the crystalline nor atomic structure. Continuum mechanics allows us to describe elastic media with effective elastic parameters, and with a constitutive law which states a linear relationship between the stress and strain, namely Hooke's law. In an elastic solid with no attenuation, when applying Newton's second law together with Hooke's law, we obtain the linear elastodynamic system of equations which can be expressed as

$$\rho u_{i,tt} = (c_{ijkl} u_{k,l})_{,j} + f_i \tag{1.1}$$

where $\boldsymbol{f}(\boldsymbol{x}, t)$ denotes the source field, $c_{ijkl}(\boldsymbol{x})$ the elastic parameters, which are components of the stiffness tensor, $\rho(\boldsymbol{x})$ the density, and $\boldsymbol{u}(\boldsymbol{x}, t)$ the displacement field. Here, I use the Einstein convention for summation over indices i, j, k, l that take value in $\{1, 2, 3\}$, and a comma between subscripts to indicate space and time partial derivatives.

In a general elastic solid, the stiffness tensor has 21 independent elastic parameters. In an isotropic solid, many of these parameters must be equal or zero due to the symmetries of the system. A full analysis (e.g. Slawinski, 2003) shows that only two independent parameters, for instance the so-called Lamé parameters λ and μ, are necessary to fully describe the propagation in the solid. The analysis of the resulting linear system in terms of eigenvalues exhibits two wave modes propagating in the elastodynamic system:

- Compressional P waves, of velocity $V_P = \sqrt{\frac{\lambda+2\mu}{\rho}}$, implying a displacement vector in the same direction as the wave propagation;
- Shear S waves, of velocity $V_S = \sqrt{\frac{\mu}{\rho}}$, implying a displacement vector in the plane orthogonal to the propagation direction. We always have $V_S < V_P$ in Earth materials.

In a fluid, where S waves do not propagate, or in a solid where we consider only P waves, if we now consider the pressure field $w(\boldsymbol{x}, t)$ in a medium with constant density, we obtain the scalar acoustic wave equation

$$\frac{1}{c^2} w_{,tt} - w_{,ii} = f, \tag{1.2}$$

where $c(\boldsymbol{x})$ denotes the compressional wave speed and $f(\boldsymbol{x}, t)$ is the volumetric distribution of pressure source divided by the compressibility of the medium.

These partial differential wave equations (PDE) cannot be solved analytically in arbitrary heterogeneous media. Therefore, numerical methods must be employed to discretize the partial derivatives in time and space and solve the partial differential equations. The most common methods for that purpose are finite-difference methods (FD), spectral and pseudo-spectral methods, and continuous or discontinuous finite-element methods (FEM/DG)(Virieux et al., 2011).

FD methods estimate derivatives by the use of a Taylor expansion. They have been widely used for solving wave equations for their simplicity and efficiency (Virieux, 1986a; Moczo et al., 2004; LeVeque, 2007; Taflove and Hagness, 2005).

The other numerical methods aim at building a numerical solution in a subspace as a linear combination of basis functions with associated weights (degrees of freedom).

- In the spectral and pseudo-spectral methods, the solution is expanded in a suitable set of global basis functions such as plane waves in the spatial Fourier domain. These methods make intensive use of fast Fourier transform (Fornberg, 1996).
  - In the spectral method for boundary integral equations, the PDE is solved in the spectral domain, with complex boundary conditions.
  - In the pseudo-spectral method, the PDE is solved in the configuration space.
- Continuous FEM rely on a piecewise discretization of the spatial domain. The solution is expanded in a set of local basis functions at nodes located inside the cells. Some of those nodes might be shared between cells to enforce continuity of the global solution (Zienkiewicz and Morgan, 2006; Marfurt, 1984).
- As a subset of the latter, spectral-element methods (SEM) are high-order continuous FEM which make use of Legendre polynomial basis functions and Gauss-Lobato-Legendre quadrature rules in order to optimize the convergence rate (see for instance Komatitsch and Vilotte, 1998; Trinh et al., 2018).
- Finally, in the DG formulation, none of the nodes are shared between cells. Neighboring cells are connected by an additional integral term in the formulation, which expresses a numerical flux between cells (Reed and Hill, 1973; Käser and Dumbser, 2006; Brossier, 2009; Bonnasse-Gahot et al., 2018). The DG method can be seen as a high-order generalization of the finite-volume (FV) method, where the finite element approximation is piecewise constant inside elements LeVeque (2002).

Continuous and discontinuous FEM generally handle boundary conditions in a natural way thanks to the integral formulation, and they provide flexibility in terms of geometry with the use of deformed meshes.

Wave equations govern highly-oscillating wavefields like pressure or displacement. Despite the advantage of linearity of these equations, it is worth noting that the simulation of the wave propagation over hundreds or thousands of wavelengths might be extremely demanding in terms of computational resources, because of the spatial and temporal discretization required to keep an acceptable accuracy. Generally, 5 to 10 points per wavelength are required in the spatial grid for a good representation of the signal, while a Courant–Friedrich–Lewy (CFL) condition imposes a strong constraint on the time step in relation with the spatial grid spacing when solving explicit systems. This yields a large number of degrees of freedom, and, consequently, a heavy computational burden.

In this context, if the medium properties are smoothly varying, we might expect the wave behavior to be smoothly affected by the medium, in comparison with the wave oscillation frequency. In other words, we could assume a scale separation between the wave oscillation and the variation of quantities like its direction of propagation, its velocity, its amplitude etc. This assumption is behind the high-frequency hypothesis, and it has led to abundant theoretical and numerical developments. In what follows, I give some theoretical details together with the fundamental equations obtained under this hypothesis.

1.1.2 A high-frequency approximation in inhomogeneous isotropic media

1.1.2.1 Eikonal and transport equations

We first convert the scalar wave equation (1.2) to the frequency domain by applying a time Fourier transform. For an angular frequency ω, we set $w(\boldsymbol{x},t) = v(\boldsymbol{x},\omega)\exp(\mathrm{i}\omega t)$. Then, v satisfies the Helmholtz equation

$$c^2\Delta v + \omega^2 v = 0. \tag{1.3}$$

We might try to find a solution under the form of the following ray ansatz in the frequency domain (Engquist and Runborg, 2003):

$$v(\boldsymbol{x},\omega) = \mathrm{e}^{\mathrm{i}\omega\phi(\boldsymbol{x})}\sum_{k=0}^{\infty} A_k(\boldsymbol{x})(\mathrm{i}\omega)^{-k}. \tag{1.4}$$

Here, $A_k(\boldsymbol{x})$ denotes the amplitude coefficients and $\phi(\boldsymbol{x})$ the phase function. This ansatz is known as the Wentzel–Kramers–Brillouin (WKB) expansion. Phase and amplitude are assumed to be smooth functions. The separation of scales between a highly oscillatory part (exponential term with a large ω) and smoothly varying phase and amplitude makes the ansatz a suitable candidate for describing high-frequency wave propagation in smoothly inhomogeneous media. For large ω, only the first term A_0 is significant in the ansatz, yielding a solution of the form (Červený, 2001; Virieux and Lambaré, 2015)

$$v(\boldsymbol{x},\omega) = A_0(\boldsymbol{x})\mathrm{e}^{\mathrm{i}\omega\phi(\boldsymbol{x})}. \tag{1.5}$$

One can recognize that in the ansatz, the equation $\phi(\boldsymbol{x}) = t$ represents the moving wavefront at time t. We then interpret the evolution of the phase function between two successive wavefronts as the propagation time of the wave (see Engquist and Runborg, 2003). Therefore, assimilating the phase function $\phi(\boldsymbol{x})$ with traveltime $T(\boldsymbol{x})$, and introducing this simple ansatz (1.5) into the Helmholtz equation (1.3), we obtain

$$\omega^2 A\left[(\nabla T)^2 - \frac{1}{c^2}\right] + \mathrm{i}\omega\left[2\nabla A\cdot\nabla T + A\nabla^2 T\right] + \nabla^2 A = 0. \tag{1.6}$$

Here, I have dropped the zero index so that $A_0(\boldsymbol{x})$ is simply denoted by $A(\boldsymbol{x})$ in what follows. Expression (1.6) must be satisfied for any frequency ω. Since we are in the high frequency regime, we focus on the terms in ω^2 and ω, and we neglect the third one. We obtain two fundamental partial differential equations, the Eikonal equation,

$$\|\nabla T\|^2 = \frac{1}{c^2}, \tag{1.7}$$

and the transport equation,

$$2\nabla A\cdot\nabla T + A\nabla^2 T = 0. \tag{1.8}$$

This asymptotic approximation obtained in the limit $\omega \to \infty$ is related to what is called ray theory, or geometrical optics theory when considering the propagation of light. Indeed, more terms from the A_k expansion could be considered. Besides, the third term in equation (1.6) could be taken into account for more precise approximations in finite frequency. It could be involved as a source term in the right-hand side of the transport equation, or combined with the first term to form a frequency-dependent Eikonal (Biondi, 1992; Zhu and Chun, 1994). Moreover, ansatz (1.4) is one simple expression chosen among a large variety of expansions. In section 1.2.1, we will see that more elaborate ansatz might be chosen in order to account for physical phenomena that are not included in this simple ansatz. All these alternatives provide correction terms to the Eikonal and transport equations. However, I do not consider these extensions in my work and I stick to the infinite frequency limit, which is an appropriate approximation in many applications.

Since $\phi(\boldsymbol{x}) = t$ in the ray ansatz represents the moving wavefront at time t, wavefronts are perpendicular to the phase gradient or equivalently the traveltime gradient. The traveltime gradient $\nabla T(\boldsymbol{x})$ is called the slowness vector, and denoted by $\boldsymbol{p}$. Curves carrying the energy flux are called rays. In isotropic media, rays are parallel to the slowness vector field. There is an intuitive way of understanding equations (1.7) and (1.8). The traveltime from a point source to a given point $\boldsymbol{x}$ depends on the ray geometry (the path of the energy flux) between the source and $\boldsymbol{x}$; so does the local direction of propagation at point $\boldsymbol{x}$. In other words, we cannot know a priori the direction of propagation at point $\boldsymbol{x}$. However, locally, we know the wave speed value $c(\boldsymbol{x})$. Thus, even without a priori knowledge of the local direction of the ray traveltime gradient $\nabla T(\boldsymbol{x})$, we know that its norm must verify equation (1.7). The Eikonal equation means nothing but stating, in an Eulerian point of view, that we assign a local speed to a given propagative phenomenon, being light or seismic wave, in the high frequency approximation. We see that the non-linearity of the Eikonal equation, expressed by the norm of the gradient, comes directly from the fact that the local propagation direction is not known. The transport equation also have an intuitive meaning. It is a pure advection equation for the energy, related to the square of the amplitude, flowing along rays. Consider a ray tube, which is formed by small perturbations around a ray. Writing the energy conservation between two successive times and applying the divergence theorem over an infinitesimal volume of the ray tube yields the local transport equation, as shown in Virieux and Lambaré (2015) (see figure 1.1). The amplitude variation is due to the geometrical spreading, which expresses how rays converge or diverge, and to the variation of speed in the medium.

The Eikonal equation (1.7) is a non-linear partial differential equation of the first order for traveltime, and it belongs to the Hamilton–Jacobi family, under the general formulation

$$\mathcal{H}\left(\boldsymbol{x}, \nabla T\left(\boldsymbol{x}\right)\right) = 0. \tag{1.9}$$

Once the traveltime is known from solving the Eikonal equation, the transport equation (1.8) is a linear partial differential equation of the first order for amplitude. It can be written in terms of amplitude squared as a conservation law, stating

$$\nabla \cdot (A^2 \nabla T) = 0, \tag{1.10}$$

which reveals the underlying energy conservation.

Eikonal and transport equations must be complemented with suitable boundary conditions, such as omnidirectional point-source conditions. These conditions will be discussed in chapter 2 for the Eikonal equation and in chapter 4 for the transport equation.

More details on the infinite-frequency limit and geometrical optics from a computational point of view can be found in the literature, as well as numerous developments to take into account singularities

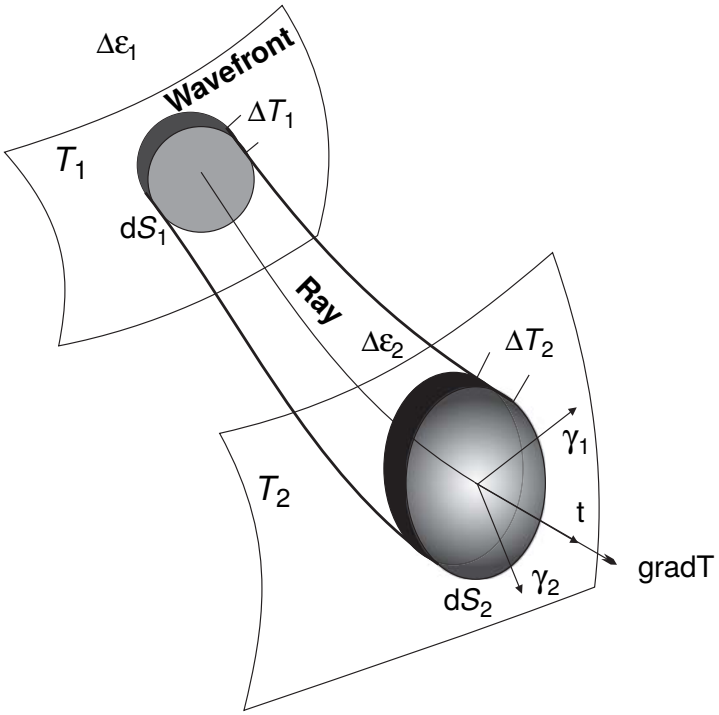


Figure 1.1: Conservation of energy along a ray tube: The energy $A^2 \mathrm{d}S\Delta T$ over a small volume of length ΔT crossing the wavefront T through the surface $\mathrm{d}S$ must be the same at times T_1 and T_2. This yields the local transport equation (from Virieux and Lambaré, 2015)

.

and finite-frequency effects, see for instance Benamou (2003); Engquist and Runborg (2003); Runborg (2007); Virieux and Lambaré (2015).

1.1.2.2 Links between traveltime and amplitude, and the wavefield

The ray ansatz (1.5) gives a direct link between an asymptotic wavefield and the traveltime and amplitude quantities. In a medium of given velocity $c(\boldsymbol{x})$, it is possible to reconstruct an asymptotic wavefield after solving Eikonal and transport equations. It is worth noticing that Eikonal and transport equations can be solved rigorously in arbitrary heterogeneous media, without violating the high-frequency hypothesis. This is because the angular frequency ω does not appear in these equations. However, the high-frequency hypothesis has to be verified as soon as the ansatz (1.5) is used to propagate a given frequency. The wavelengths introduced must be small with respect to the medium typical wavelengths.

If several solutions corresponding to several seismic waves (P wave, S wave, first arrival, later arrivals...) have been computed at a given point $\boldsymbol{x}_r$ thanks to the asymptotic theory, it is then possible and straightforward to sum the solutions given by the ansatz for each wave and obtain, after an inverse Fourier transform, a synthetic seismogram at $\boldsymbol{x}_r$, that is, the time evolution of the pressure or displacement field at $\boldsymbol{x}_r$. When reconstructing a displacement field in an elastic medium, the polarization of the waves are required. The polarization gives the direction of the oscillating displacement induced by the wave. In an elastic isotropic medium, this direction for P waves is the same as the direction of propagation. For S waves, the polarizations satisfy differential equations, detailed in Červený (2001, section 6.4) and Chapman (2004, p. 181) for instance.

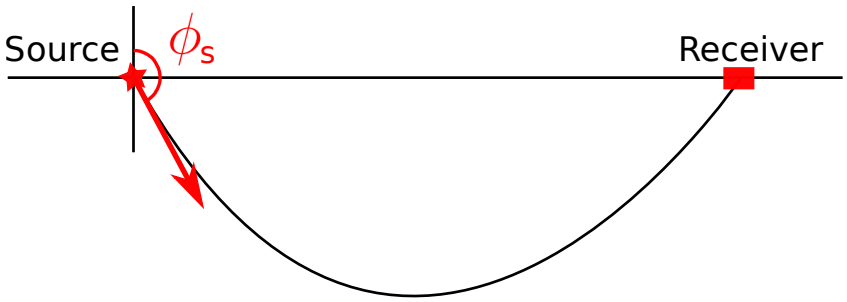


Figure 1.2: Illustration of the take-off angle $\phi_{\boldsymbol{s}}$ for a seismic ray between a source and a receiver at the surface. In the Eulerian viewpoint, the ray is a level set of $\phi_{\boldsymbol{s}}$.

This seismic wave separation/summation is valid only at infinite frequency. In practice, finite-frequency effects imply interactions between waves in heterogeneous media.

On the opposite, the high-frequency quantities, traveltime and amplitude, might be retrieved from the observed full wavefields. This has been done for traveltime since the first seismic observations were recorded, either manually or in an automated way. Traveltime and amplitude might be estimated by picking, as long as a specific wave type can be identified, separately from others, in a seismogram (see the seismogram in figure 3). They also might be obtained by cross-correlation techniques.

1.1.2.3 An equation for the take-off angle

In the high-frequency limit, we have defined rays as the curves carrying the energy flux, and the associated slowness vector $\boldsymbol{p}$ pointing in the ray direction. In two dimensions, if we consider an omnidirectional source point $\boldsymbol{x_s}$, rays are emitted in all the directions from that point and travel in the medium. In practice, for a given ray away from the source, it might be interesting to know what was the initial direction of the ray at the source. We then define, at every point $\boldsymbol{x}$ of the medium, the take-off angle $\phi_{\boldsymbol{s}}(\boldsymbol{x})$ as the shooting angle, at the source $\boldsymbol{s}$, which yields the ray linking the source $\boldsymbol{s}$ and the considered point $\boldsymbol{x}$ (see figure 1.2). Obviously, all the points along a given ray share the same take-off angle. Therefore, the take-off angle stays constant along a ray. In an isotropic medium, rays are perpendicular to the wavefronts. Consequently, the gradient of the take-off angle field $\nabla_{\boldsymbol{x}}\phi_{\boldsymbol{s}}(\boldsymbol{x})$ is tangent to the wavefront, and orthogonal to $\nabla_{\boldsymbol{x}}T(\boldsymbol{x})$. Hence, a simple PDE can be stated as

$$\nabla_{\boldsymbol{x}}\phi_{\boldsymbol{s}}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}}T(\boldsymbol{x}) = 0. \tag{1.11}$$

Similar to the Eikonal equation, equation (1.11) is an HJ equation of the form of equation (1.9), for the unknown field $\phi_{\boldsymbol{s}}(\boldsymbol{x})$, with $\nabla_{\boldsymbol{x}}T(\boldsymbol{x})$ being known from a prior traveltime computation. Unlike the Eikonal equation, the take-off angle equation is linear. To give a complete definition, boundary conditions must be added to equation (1.11). A suitable point-source condition should describe the behavior of $\phi_{\boldsymbol{s}}(\boldsymbol{x})$ in the vicinity of the source $\boldsymbol{s}$. I may write this condition in polar coordinates (r, θ)

$$\phi_{\boldsymbol{s}}(r, \theta) = \theta \tag{1.12}$$

for an arbitrarily small $r \neq 0$.

The theoretical extension to three dimensions is straightforward. It involves two angles, the dip and azimuth angles, which are also conserved along a ray, yielding two angle equations, each one similar to equation (1.11), one for each angle.

1.1.3 Conclusion: Why do we need a high-frequency approximation?

The high-frequency approximation yields two PDEs, namely nonlinear Eikonal equation and linear transport equation. These equations give the relationship between the wave speed in a given medium and the traveltime and amplitude, respectively, of waves that propagate within this medium. Inside this framework, we can also state a PDE for the take-off angle. The underlying assumption of the high-frequency approximation is that the propagation wavelength is significantly smaller than the typical scale of variation of the medium properties. It has already been mentioned that propagating hundreds or thousands of wavelengths in a full-wavefield simulation can be extremely costly, so that the high-frequency framework, when applicable, is advantageous. This is why traveltime, amplitude, and take-off angle are used in plenty of geophysical applications. A non-exhaustive list is given here.

- *Earthquake location*: In seismology, the problem of finding accurate locations for earthquakes relies on picking arrival times in seismograms from a set of stations. Given a model of the Earth's interior, which could be simple (e.g. homogeneous or layered) or more elaborate (e.g. tomographic model) depending on the amount of information available, the best location is the one for which computed traveltimes best fit with picked traveltimes. Generally, the misfit is measured in ℓ^1 or ℓ^2 norm (Lee and Lahr, 1975; Klein, 1978; Nelson and Vidale, 1990).

- *Focal mechanism inversion*: In seismology, the focal mechanism of an earthquake describes the deformation around the source that generates the seismic waves (Wilson, 1965; Sykes, 1967). Focal mechanism is a useful tool for understanding fault zones. The take-off angle is required for the computation of focal mechanisms. For shallow microseismic events, the take-off angle determination must be performed with a high accuracy.

- *First-arrival tomography in seismology*: Traveltime tomography is the main method for the determination of the long-wavelength seismic velocity structure of the Earth (Aki and Lee, 1976; Rawlinson and Sambridge, 2003; Thurber and Ritsema, 2007). Considering a large set of earthquakes, the tomographic problem consists of finding a velocity model that best explains the data. In first-arrival tomography, the data consists of a set of first-arrival traveltimes picked in seismograms. A ℓ^2 misfit function is formed involving the ℓ^2 norm of the difference between observed and computed traveltimes. The process begins with a simple velocity model, and model update is then computed iteratively using optimization methods in order to minimize the misfit function value. For this non-linear process, regularization terms must be considered with care (see e.g. Rawlinson et al., 2014). In seismology, a relocation of earthquakes is performed in the new model at each update.

- *Tomography for exploration geophysics*: Tomography is not only used in seismology, but also in exploration geophysics. Controlled sources are used instead of earthquakes. Sources are located at the surface, as well as receivers. Several modalities can be employed. First-break tomography in exploration geophysics (Zhu et al., 1992; Osypov, 2000; Leung and Qian, 2006; Taillandier et al., 2009; Noble et al., 2010; Li et al., 2013) is very similar to earthquake tomography, except that the source is perfectly located in time and space, and the data generally comes from refracted waves. Another modality is reflection tomography, in which the data consists of traveltimes picked from reflected phases (Bishop et al., 1985; Farra and Madariaga, 1988; Dyer and Worthington, 1988; Carrion et al., 1993). From these reflection traveltimes, the goal of reflection tomography is to determine a velocity distribution together with reflector positions. Finally, slope tomography inverts for both traveltime and slope of the reflectors at depth (Billette and Lambaré, 1998; Lambaré, 2008; Tavakoli F. et al., 2017; Sambolian et al., 2018).

- *Initial velocity model building for full waveform inversion*: Tomography aims to retrieve a smooth velocity model, often referred to as background model or reference model, which is needed for seismic imaging (migration). A good initial velocity model is also needed in full waveform inversion (FWI), an inversion technique which inverts not only for traveltimes or other information extracted from the traces (seismograms), but for the whole signal. Even if the theory was developed more than thirty years ago, FWI has become more and more popular in the recent years with the recording of large-offset datasets and the development of large enough computational infrastructures. FWI appears as a powerful technique to retrieve high-resolution information on the Earth's interior (Lailly, 1983; Tarantola, 1984; Virieux and Operto, 2009; Virieux et al., 2017).

- *Velocity analysis*: In exploration geophysics, a typical common mid-point (CMP) gather plots the traces for various source-receiver pairs sharing the same mid-point, with different offsets. Picking the arrival time on these CMP gathers and the curvature of the resulting reflection hyperbola is a way to retrieve a medium velocity above a flat horizontal interface, using the normal moveout (NMO). For dipping interfaces, the dip-moveout correction is involved (Yilmaz, 2001).

- *Kirchhoff/Born migration*: Seismic migration consists of back-projecting in depth the energy contained in a recorded reflection dataset, so as to retrieve the short-wavelength components of the subsurface velocity. Basic algorithms require traveltime and amplitude (Beylkin, 1985; Bleistein, 1987; Beylkin and Burridge, 1990; Jin et al., 1992; Gray and May, 1994; El-Mageed et al., 1997; Lambaré et al., 2003; Operto et al., 2003; Alkhalifah, 2011; Li and Fomel, 2013). More accurate formulations also involve angles at depth and at the surface, requiring to compute the take-off angle. I will give more details on these techniques in chapter 5. The recent developments of seismic migration involve full-wavefield techniques, such as in the case of reverse-time migration (RTM), where the high-frequency approximation is not used. However, I will show in chapter 5 that we can design preconditioning techniques based on asymptotic theory which could be useful in the RTM process.

- *Diffraction imaging*: While classical seismic imaging techniques rely on reflection events, diffracted waves might provide additional information for high-resolution imaging, and investigation of faults, for instance. Diffraction imaging is based on the focusing to points of primary diffractions. The separation of diffractions from reflections can be performed by the common-reflection-surface (CRS) method, in which first and second-order spatial derivatives of the traveltime are used (Dell and Gajewski, 2011). Diffractions can then be included in a wavefront tomography algorithm (Duveneck, 2004; Bauer et al., 2017).

In the next section, I detail several practical approaches for computing traveltime, amplitude, and take-off angle.

1.2 Computing traveltime, amplitude, take-off angle: state of the art

Here, I review some aspects of the numerical computation of asymptotic quantities, from various point of views. The Lagrangian point of view, for which I detail the main principles, has been historically the first one to be exploited as a method of choice for traveltime computation. I will mention briefly some so-called semi-Lagrangian approaches. Finally, I will describe with further details the Eulerian point of view, which I will employ in all the following.

1.2.1 The Lagrangian point of view: Ray tracing

A first method for computing the traveltime from a source point in the asymptotic approximation is the ray-tracing approach, which uses the classical method of characteristics (Courant and Hilbert, 1966). Until the early 80's, apart from some graphical methods based on wavefront tracing suitable for simple cases (Thornburgh, 1930; Riznichenko, 1946), this was the nearly unique existing method to compute traveltime solutions. Rather than directly solving the Eikonal PDE, a set of linear ODEs is derived which can be efficiently solved by integration following a Lagrangian approach: a seismic ray is assimilated to a particle trajectory. Let $\boldsymbol{q}$ be the current position of the particle, l the arclength along the ray, and $s(\boldsymbol{q})$ the slowness at particle position $\boldsymbol{q}$. In an isotropic medium, we follow the derivation proposed in Forgues (1996) and define the unit vector $\boldsymbol{t}$ tangent to the ray at position $\boldsymbol{q}$ by

$$\boldsymbol{t} = \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}l}. \tag{1.13}$$

In an isotropic medium, rays are perpendicular to wavefronts, so that vector $\boldsymbol{t}$ is parallel to the gradient of traveltime ∇T. From the Eikonal equation (1.7), we thus have

$$\boldsymbol{t} = \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}l} = \frac{1}{s(\boldsymbol{q})}\nabla T(\boldsymbol{q}). \tag{1.14}$$

Now, we consider the evolution along the ray of the vector ∇T tangent to the ray:

$$\frac{\mathrm{d}\nabla T}{\mathrm{d}l} = \frac{\mathrm{d}}{\mathrm{d}l}\left(s(\boldsymbol{q})\frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}l}\right). \tag{1.15}$$

From equation (1.14) we have

$$\frac{\mathrm{d}}{\mathrm{d}l} = \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}l}\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{q}} = \frac{1}{s(\boldsymbol{q})}\nabla T(\boldsymbol{q})\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{q}}. \tag{1.16}$$

Therefore

$$\frac{\mathrm{d}\nabla T}{\mathrm{d}l} = \frac{1}{s}\nabla T\frac{\mathrm{d}\nabla T}{\mathrm{d}\boldsymbol{q}} = \frac{1}{s}\nabla T\ \ \nabla(\nabla T) = \frac{1}{2s}\nabla\left(\|\nabla T\|^2\right) = \frac{1}{2s}\nabla(s^2) = \nabla s. \tag{1.17}$$

Finally, considering the slowness vector $\boldsymbol{p} = \nabla T$, we obtain the ray tracing system

$$\begin{cases} \dfrac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}l} = \dfrac{1}{s}\boldsymbol{p}, \\ \dfrac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}l} = \nabla s. \end{cases} \tag{1.18}$$

Together with the Eikonal equation (1.7), this system can be solved for given initial or boundary conditions (Julian and Gubbins, 1977; Červený et al., 1988; Virieux and Farra, 1991; Červený, 2001; Chapman, 2004).

A more general framework involves the Hamiltonian formalism of classical mechanics, using a Hamiltonian $\mathcal{H}(\boldsymbol{q}, \boldsymbol{p})$ such that setting $\mathcal{H} = 0$ satisfies the Eikonal equation. In the Hamiltonian approach, rays in the standard space are replaced by characteristic curves in the phase space ($\boldsymbol{q}$, $\boldsymbol{p}$). An independent variable t is then considered, yielding the Hamilton equations

$$\begin{aligned} \frac{\partial\mathcal{H}}{\partial\boldsymbol{p}} &= \frac{\mathrm{d}\boldsymbol{q}}{\mathrm{d}t}, \\ -\frac{\partial\mathcal{H}}{\partial\boldsymbol{q}} &= \frac{\mathrm{d}\boldsymbol{p}}{\mathrm{d}t}. \end{aligned} \tag{1.19}$$

Depending on the Hamiltonian that has been chosen, the variable t has different meanings: time, arclength, particle stepping, for instance. Considering the following Hamiltonian

$$\mathcal{H}(\boldsymbol{q}, \boldsymbol{p}) = \|\boldsymbol{p}\| - s(\boldsymbol{q}), \tag{1.20}$$

we retrieve the case derived above, with $t \equiv l$. When considering anisotropy, we will see that another formulation has a special interest, for the computation of the group velocity vector. The corresponding Hamiltonian writes

$$\mathcal{H}(\boldsymbol{q}, \boldsymbol{p}) = \frac{1}{2}\left(\frac{\boldsymbol{p}^2}{s^2(\boldsymbol{q})} - 1\right). \tag{1.21}$$

In this case, t corresponds to the traveltime $T(\boldsymbol{q})$ along the ray.

Two main categories of ray tracing schemes have been designed. In shooting methods, at a given point source, an initial trajectory of the ray, namely a given take-off angle, is defined. This yields initial position $\boldsymbol{q}$ and slowness vector (momentum) $\boldsymbol{p}$. The ray tracing system (1.19) is then integrated, thus providing a new position and slowness vector at each step. Traveltime is updated along the trajectory during the integration, through the relationship between the integration parameter and the traveltime. For instance, if Hamiltonian (1.21) is employed, the integration step directly correspond to the traveltime step. If Hamiltonian (1.20) is employed, then the integration step must be multiplied by the local slowness to obtain a traveltime step.

In practice, when looking for the traveltime between source/receiver pairs, a fan of rays is shot from the source, and the correct path and related traveltime to connect sources and receivers may then be approached iteratively, and/or involving interpolation. Alternatively, in bending methods, an initial, arbitrary, and probably non-physical path is defined between two points (a source and a receiver). This ray path is then bent by a perturbation method until it satisfies a minimum traveltime criterion based on the Fermat principle so that it becomes a physical ray. This process might not converge in a finite number of trials. More details and reviews of ray tracing schemes might be found in Virieux (1996), Rawlinson et al. (2007) and Virieux and Lambaré (2015).

Traveltime is directly accessed by integration along rays. When it comes to amplitude, an additional computation of a geometrical spreading factor is required, which expresses how rays are converging or diverging. This computation of amplitude does not need to solve the transport equation, but requires only the paraxial ray equations, coming from the first-order perturbation of the ray tracing system around a central ray, thanks to the energy flux conservation (Chapman, 1985; Farra and Madariaga, 1987). Finally, the take-off angle is directly available, since it corresponds to the initial angle in the shooting approach: the initial condition corresponds to the take-off angle. Hence, each ray is defined by this parameter.

The ray tracing method, although easy to derive and very efficient, exhibits some limitations for practical applications. For applications such as imaging or traveltime tomography for velocity model building, the ray traveltime must be interpolated between a fan of rays in order to obtain traveltime values where they are required (at receivers). However, in inhomogeneous media, ray paths might be complex, and the high non-linearity of the ray tracing problem results in a non-uniform sampling of the medium (see figure 1.3) and potential failure in joining two points with a ray in an accurate way Rawlinson et al. (2007). Therefore, interpolation might be challenging.

Moreover, when triplication occur in the wavefield, the multivaluedness of the traveltime solution results in an uncertainty on which phase has been computed by a given ray. In other words, it might be difficult to ensure that the phase computed by the ray correspond to the phase observed in the data (first arrival, most energetic arrival etc.).

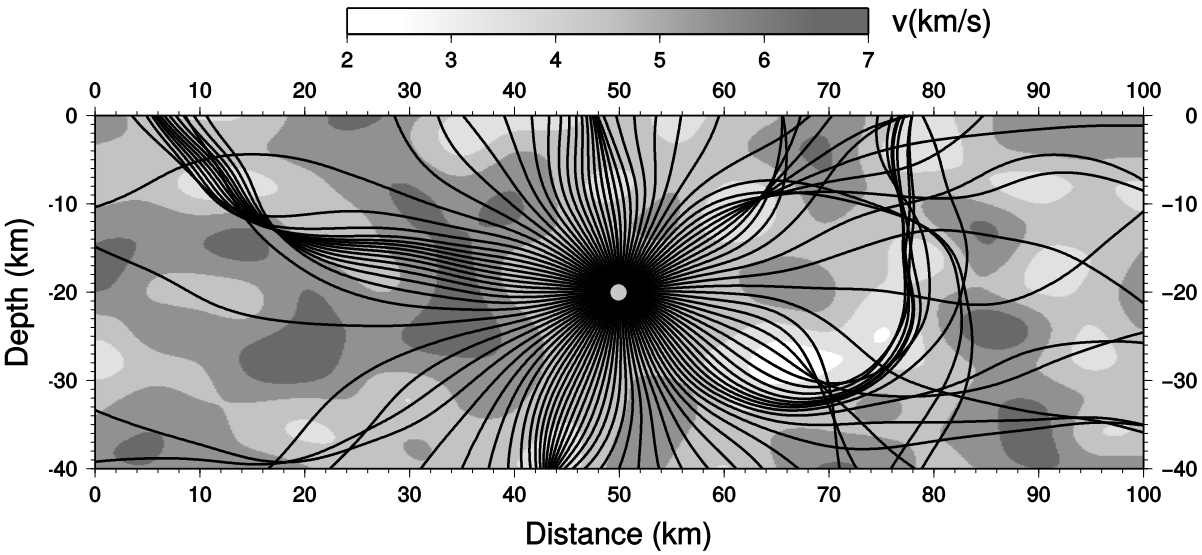


Figure 1.3: A uniform fan of rays emitted by a source point in a heterogeneous medium. The resulting sampling of the medium is non-uniform, so that traveltime interpolation is not possible in some regions. Image from Rawlinson et al. (2007).

In addition, some physical phenomena are not accounted for with classical ray theory (Engquist and Runborg, 2003). First, the amplitude computation fails at caustics, which denotes points or curves where rays cross or merge. At these caustics, the ray tube displayed in figure 1.1 collapses and the amplitude goes to infinity, so that the classical asymptotic ray method is inapplicable. The field in the vicinity of caustics was studied by Kravtsov (1964) and Ludwig (1966), and further investigated by Babic and Buldyrev (1991) with the use of sophisticated asymptotic ray expansions with Airy functions as a remedy. Another way to tackle caustics can be performed through Gaussian beams (Ralston, 1982; Popov, 1982), where the neighborhood of a ray is considered. The solution is constructed with a complex phase function away from the ray. This is sometimes referred to as complex geometrical optics. Another typical phenomenon is the diffraction, accounting for effects of geometry and boundary conditions. This phenomenon is lost in the geometrical optics approximation, where shadow zones may occur. Keller (1962) introduced a technique, namely the geometrical theory of diffraction (GTD), for adding this effect with extra terms in the ansatz. More precisely, expansions in integer and non-integer powers of ω are added to ansatz (1.4). Further developments were pursued later, considering various geometries, such as in Kouyoumjian and Pathak (1974). Diffraction at the tip of a wedge yields an infinite set of diffracted rays in all directions. Diffraction by scatterers such as a cylinder generates creeping rays that follow its surface and emit surface-diffracted rays behind the obstacle. Some mathematical justifications of Keller's work are given in Babič and Kirpičnikova (1979), Babic and Buldyrev (1991), and Klem-Musatov (1995).

In conclusion, the standard ray theory presents practical drawbacks. Some of them might be corrected thanks to more elaborate ray expansions. This topic has been an active and fruitful field of theoretical research. However, it is worth mentioning that in practice, these theories are very difficult to use for realistic applications because of the complexity of implementation and their inadequacy for modeling arbitrary media. For example, GTD needs geometrical description of boundaries and obstacles, which are not well defined in practice inside the Earth. Next, I present approaches that differ from the Lagrangian one. They try either to build a wavefront (semi-Lagrangian methods), or to directly

solve the Eikonal PDE in an Eulerian way.

1.2.2 Semi-Lagrangian methods

Another approach to compute the traveltime consists of sampling a wavefront and make it evolve step by step. These methods are referred to as semi-Lagrangian, since they involve Lagrangian elementary steps like ray tracing, to build not only rays, but a succession of elementary cells or segments defined by adjacent rays and forming wavefronts. This approach relies on the Huygens's principle, assuming that each point of a wavefront at time t acts like a secondary source to build the wavefront at time $t + \delta t$.

A semi-Lagrangian approach was designed by Qin and Schuster (1993), involving five steps in the numerical algorithm:

1. **Source point treatment**: Assuming a locally homogeneous medium, the source point is propagated to the first wavefront along the ray velocity directions for a suitable time step. Coordinates of the first wavefront are computed and stored at some specific discrete points.

2. **Plane-wave segment approximation**: Each discrete point of the wavefront is considered as the midpoint of a local plane segment tangent to the wavefront, and the normals to these local plane segments are computed.

3. **Propagation of the plane-wave segments**: Each segment is propagated along the ray direction to the next time step.

4. **Refining of the wavefront**: Two adjacent points might diverge or converge during the propagation step. Therefore, some points might be added of removed in order to keep a reasonably homogeneous sampling of the wavefront.

5. **Converting the traveltime to regular grids**: A linear interpolation step yields the computation of traveltime at grid points from the current wavefront.

This scheme from Qin and Schuster (1993) is designed to compute first-arrival traveltime.

Sun (1992), Vinje et al. (1993), and Lambaré et al. (1996) have proposed other approaches, based on the ray tracing equations, in order to compute simultaneously traveltime and amplitude thanks to the paraxial ray equations. Multivaluedness of the traveltime solution has been considered (Vinje et al., 1996b,a), so that complex folded wavefronts can be tracked, including caustics. Ray density is controlled by a sampling criterion, in order to obtain a uniform sampling of the ray field in the phase space.

The principles of the semi-Lagrangian methods are attractive because they offer a way to build complex wavefronts and refine them to a given level of accuracy, and they avoid shadow-zone issues. They are the current working horse in the NORSAR software (`https://www.norsar.no/software/norsar-3d/`). However, in terms of numerical implementation, these methods imply a quite complex machinery, including back and forth conversions between the Lagrangian ray paths and the Eulerian grid points at each step by means of interpolation procedures. They also imply a large amount of bookkeeping for the geometry of the wavefront, which can have a high complexity.

1.2.3 Eulerian methods

Introduction Ray tracing is a way to compute traveltime and amplitude in a Lagrangian viewpoint, considering one particle with a given shooting parameter (take-off angle) and tracing its path inside the medium. However, as discussed previously, for practical applications in complex media, ray tracing might fail in some parts of the medium. Therefore, ray tracing might not be the most suitable tool for applications which require traveltime values at many points of a given grid. The semi-Lagrangian methods combine the Lagrangian point of view with the Eulerian point of view of computing wavefronts and quantities at grid points. However, they imply complex numerical strategies. It is therefore essential to consider the pure Eulerian point of view, where the principle is to solve Eikonal, transport, and take-off angle partial differential equations themselves instead of deriving a system of Lagrangian ordinary differential equations.

In what follows, I will mainly focus on the Eikonal equation since its non-linearity makes it the most challenging one. Investigations on solving the Eikonal PDE in a geophysical context have started to attract attention in the early 1990's (Vidale, 1988a; van Trier and Symes, 1991; Podvin and Lecomte, 1991; Hole and Zelt, 1995). Numerous solvers have been proposed until today, a large majority of them relying on the finite-difference method. Yet challenges remain, as shown at the end of this chapter. Here, I give an overview of the development of these Eikonal solvers and the numerical recipes they use.

Theoretical description Formally, the Eikonal equation is non-linear. Consequently, numerical grid-based schemes for solving it are complex, and their convergence properties are generally not easy to establish. The Eikonal equation arises not only in geophysics but in many other fields of research such as geometrical optics (Benamou, 2003), optimality problems with shortest/geodesic path calculation (Moser, 1991; Kimmel and Sethian, 1998), computer vision problems with shape-from-shading (Rouy and Tourin, 1992; Kimmel and Sethian, 2001), fluid mechanics (Lifschitz and Hameiri, 1991), so that the applied mathematics community has been intensively investigating on how to solve Eikonal equations, and more generally Hamilton–Jacobi equations, in a robust way. Since classical solutions, with strong differentiability properties, do not exist in general for the global nonlinear first-order Hamilton–Jacobi equation, generalized solutions have been looked for, which satisfy the equation almost everywhere (Lions, 1982). In this context, a fundamental result from Crandall and Lions (1983) states the existence and uniqueness of a very special kind of solution among all the numerous generalized solutions, namely the viscosity solution[1]. Interestingly, this viscosity solution has a physical meaning. In the case of the Eikonal, it corresponds to the shortest path, also known as the first-arrival traveltime. A proof of this result can be found, for instance, in Benamou (1999), using the optimal control theory.

Next, I try to give a description of this viscosity concept in the Eikonal case. Although not mathematically rigorous (in particular, I do not detail the mathematical spaces in which the solutions lie), this description highlights some intuitive understandings of this viscosity concept. First, the existence of such a solution stated by Lions (1982) and Crandall and Lions (1983), and the existence of numerical approximations that converge towards this solution (Crandall and Lions, 1984) ensure that a traveltime value will be assigned at every point of the medium, including diffractions in the sense of the Huygens' principle (see e.g. von Westenholz, 1977). Consequently, a numerical Eikonal computation should be able to correctly handle shadow zones, head waves, and physical obstacles. In other words, the Eikonal

[1]The *viscosity* denomination comes from the vanishing viscosity method employed in the proof of existence of such a solution in Crandall and Lions (1983).

PDE contains more physics than the ODEs derived from it in the method of characteristics (Lagrangian ray theory).

Second, the uniqueness of the viscosity solution, proven in Crandall and Lions (1983), has a subtle meaning which is linked to the causality properties of the Eikonal. By causality, I refer to the fact that the time should always increase along a characteristic, reflecting the underlying causal wave propagation phenomenon. I show in figure 1.4 that even in a very simple 1D medium, an infinity of Eikonal solutions exist. However, in the unique viscosity solution, information flows in a causal way, filling the medium from the source. The information flow is given by the gradient of the solution. Given a point in the medium, the information coming from the source to this point flows along a strictly growing traveltime path. These paths, akin to rays in the Lagrangian point of view, are called characteristics of the Eikonal equation. Outside of the source, local minima cannot exist in the solution.

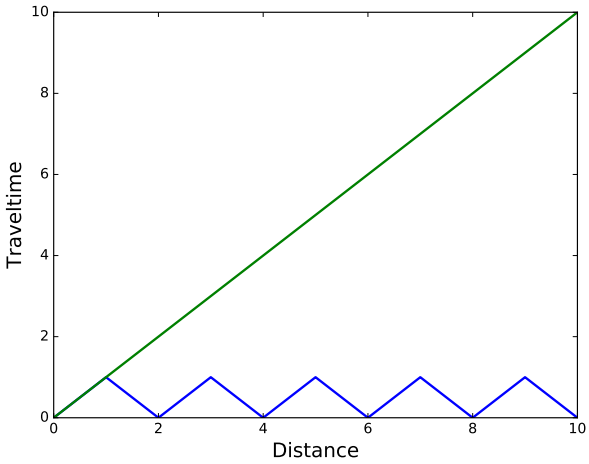


Figure 1.4: 1D homogeneous medium with velocity $c = 1$ and a point source at $x = 0$. Both blue and green solutions locally verify the Eikonal equation, that is, the local slope of the curve is $1/c = 1$. However, the blue one does not verify causality: the path from the source along x does not correspond to increasing traveltime, and there are local minima. Therefore, the green solution is the only viscosity solution.

The viscosity solution of the Eikonal, namely first-arrival traveltime $T(\boldsymbol{x})$, is a continuous function of space. This comes directly from the viscosity property: if a jump in the solution exists, then it should be filled from the lowest (first-arrival) side to retrieve the viscosity solution. This is illustrated in figure 1.5. Indeed, we assume that the velocity is defined everywhere and strictly positive.

However, the continuity does not hold for spatial derivatives, as illustrated in figure 1.6. In the 2D case of a triplication of the underlying wavefield due to a negative perturbation of the velocity in a homogeneous background medium, two branches will develop in the solution, corresponding to the circumvention of the singularity on both sides. When these two branches meet behind the velocity perturbation, a singularity will appear along the locus of same traveltime for the two branches. Along this singularity, the solution is continuous but its gradient is not. In other words, the information flows in different directions in the two branches.

I give a last illustration of the viscosity solution in figure 1.7. Here, a negative perturbation of the velocity in a homogeneous background is simulated, no longer like a Gaussian anomaly, but like a rough obstacle forming a circular discontinuity in the velocity model. The viscosity solution fills the

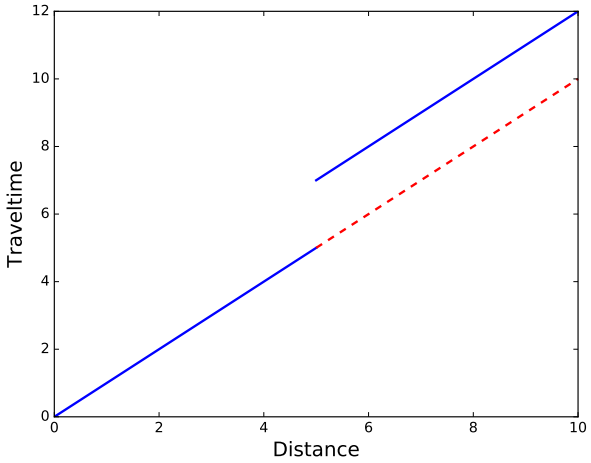


Figure 1.5: 1D homogeneous medium with velocity $c = 1$ and a point source at $x = 0$. The blue solution locally verifies the Eikonal equation. However, it does not correspond to the minimal traveltime. The continuation of the left part of the blue solution by the red dashed line results in a continuous, first-arrival solution.

obstacle with traveltime information coming from all around it, which corresponds to the first-arrival path. In such a case, the standard Lagrangian ray-tracing approach would fail: more elaborate theories would be necessary, handling above-mentioned physical phenomena like diffraction and creeping rays (Keller, 1962; Runborg, 2007).

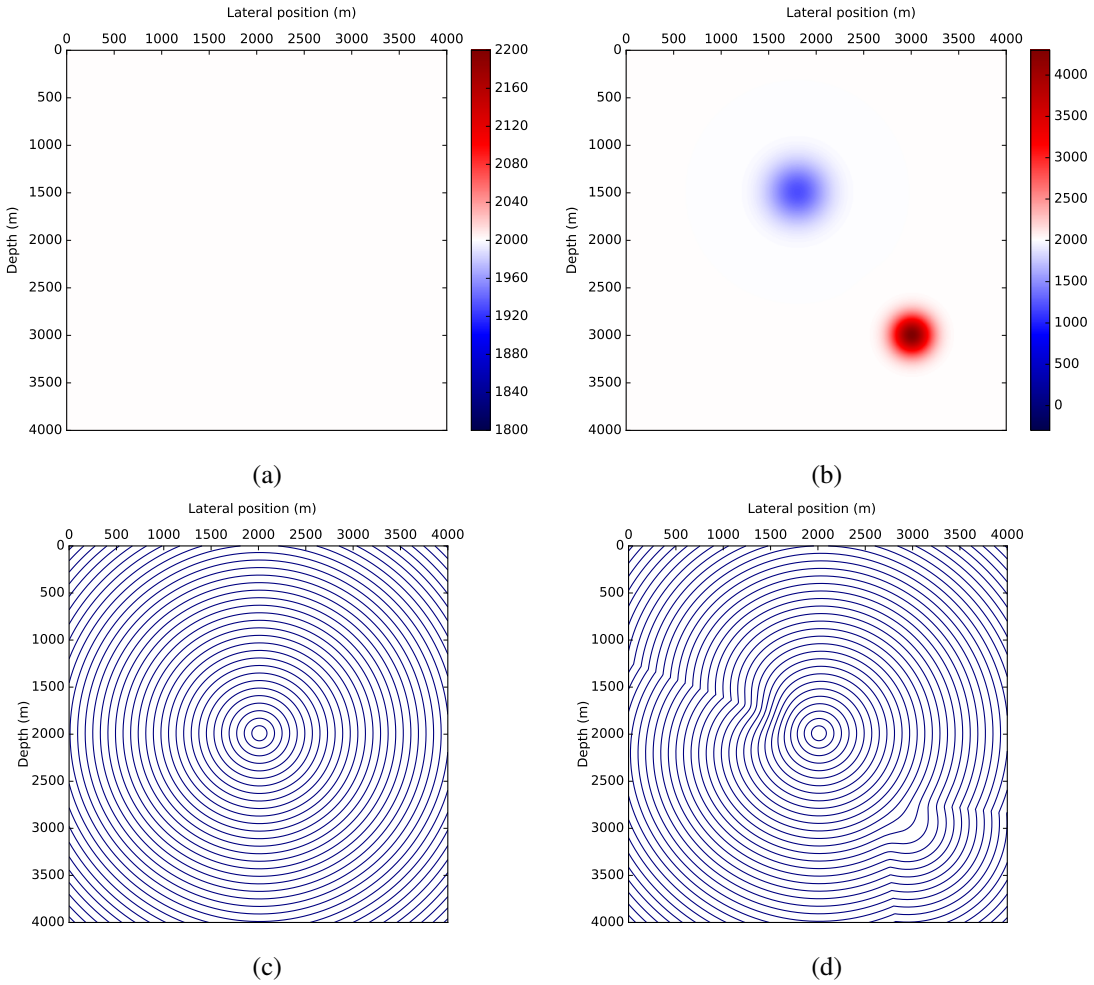
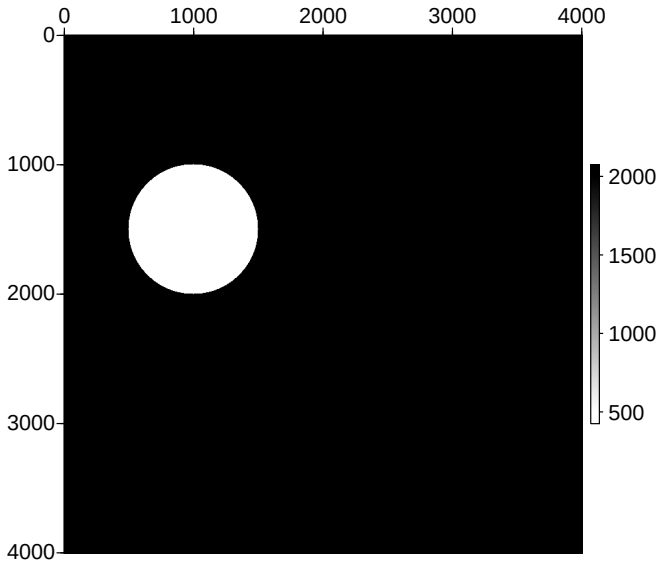
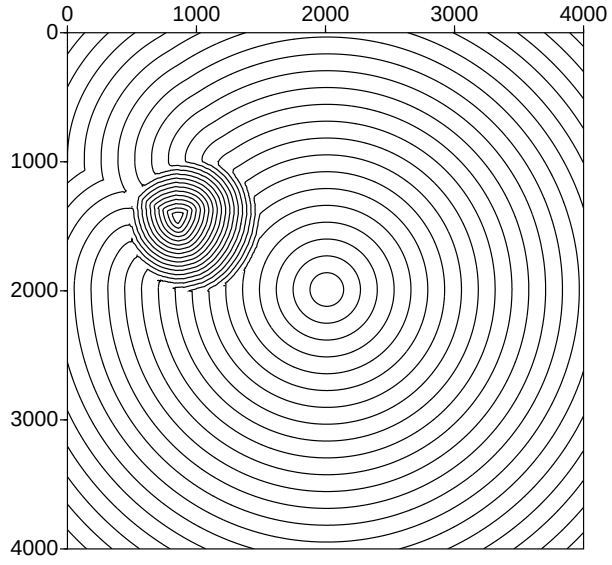


Figure 1.6: (a) 2D homogeneous background model with velocity $c = 2000m/s$, and (b) model with negative and positive velocity perturbations. A point source is located at the center of the domain. (c) Curves of iso-traveltime (akin to wavefronts) computed in a homogeneous model are circles centered around the source. The information flow is oriented towards the normal of these circles, i.e. the traveltime gradient. (d) In presence of velocity perturbations, the traveltime field is more complex. In particular, the solution behind the negative anomaly is reconstructed with information coming from both sides of the anomaly (left and right circumventions). These two circumventions result in two branches of the solution that meet along a line behind the anomaly, where they yield equal traveltime. This line singularity corresponds to a continuous traveltime solution but a discontinuity of the traveltime gradient.



(a)



(b)

Figure 1.7: (a) 2D homogeneous background model with velocity $c = 2000m/s$, and negative velocity perturbation: circular obstacle with velocity $c^\star = 500m/s$. (b) Curves of iso-traveltime for a point source located at the center of the domain. The viscosity solution, i.e. the first-arrival traveltime, propagates along the interface and penetrates inside the obstacle.

Numerical developments The numerical design of Eikonal solvers relies on two main ingredients. The first one is a local solver, or local stencil. Based on the predefined spatial discretization of the domain (generally FD), it states how the solution is computed at a given discrete point, given traveltime values at the neighbors. To do so, the Eikonal equation is discretized, and more precisely, the spatial derivatives of the solution are written with respect to neighbor values. Various number of neighbors might be involved, depending on the order of the discretization of the equation. Then, the local stencil might involve several expressions of these derivatives, and choose a preferred one based on criteria such as minimum time. At this local level, causality should be handled carefully: the information flow must be consistent. This might be expressed in different ways depending on the stencil.

The second ingredient is a global strategy which defines how the discrete points are updated. This can be performed with a pre-defined ordering, a dynamic sorting of the points depending on their traveltime values, an iterative process based on repeated sweeps of the whole grid or repeated updates of the points all at once, etc. Algorithms can be single-pass: each point will be updated only once and assigned with its final traveltime value, or multi-pass: points might be updated several times during the Eikonal computation.

In the geophysics community, Vidale (1988b) proposed an Eikonal solver in 2D, and later extended it to 3D in Vidale (1990). The principles in 2D are the followings: The velocity model is sampled at points on a regular grid with equal horizontal and vertical spacing h. Traveltime is computed at the same grid points. Traveltime derivatives are approximated using centered finite differences as follows. Inside the square defined by 4 grid points with x- and $z-$indices (i, j), $(i+1, j)$, $(i, j+1)$ and $(i+1, j+1)$, we have the following approximation

$$T_{,x} \simeq \frac{1}{2h}(T_{i\,j} + T_{i\,j+1} - T_{i+1\,j} - T_{i+1\,j+1}), \tag{1.22}$$

and

$$T_{,z} \simeq \frac{1}{2h}(T_{i\,j} + T_{i+1\,j} - T_{i\,j+1} - T_{i+1\,j+1}). \tag{1.23}$$

From these expressions, a local stencil (building block) defines how to compute the time at the fourth corner of such a square, given the times at the other three corners, using the discretized derivatives in the Eikonal equation. For example, we might write

$$T_{i+1\,j+1} = T_{i\,j} + \sqrt{2(hs)^2 - (T_{i\,j+1} - T_{i+1\,j})^2} \tag{1.24}$$

for a plane-wave approximation, where s is the slowness, i.e. the inverse of the speed. Another formula is derived in Vidale (1988b) for a circular wavefront approximation. The global strategy consists of progression of the solution by solving square rings of increasing radius around the source point. Given the traveltime in a squared surface around the source, the times are then computed at grid points adjacent to each side of the square, the surface is expanded by one grid cell, and the process is repeated. Each point is visited only once: it is a single-pass algorithm. The computation of traveltime for points located in a given side of the square is ordered based on finding a local minimum in traveltime on this side, computing the traveltime value at this first point with the plane-wave approximation (1.24), then propagating the solution to the other points of the same side using the local stencil. The initialization is performed as follows. The source (where traveltime equals to zero) is located at a grid point. Traveltime is computed at the four adjacent points by multiplying the grid spacing by the mean of slownesses at source and adjacent point. Then, the local stencil is used to complete the first ring.

This algorithm, being among the first to spread in the geophysics community, suffers from several drawbacks. In this strategy, due to the orientation of the approximation of the derivatives, the information is implicitly assumed to come from the inside of the expanding box. Hence, only the outgoing part

of the solution is modeled. This algorithm might miss some first-arrival information, like in presence of a low-velocity anomaly, and give a result that is not the viscosity solution (Hole and Zelt, 1995; Engquist and Runborg, 2003). Moreover, locally plane waves are assumed, except when circular-wavefront approximation formula is involved, resulting in a hybrid scheme, more complex, and which needs to be tuned by the user. Finally, Vidale's method requires an ordering of the grid points on each side of the current ring before computing the values for the next ring. Besides resulting in an additional step in the process, this ordering prevents parallelization of the algorithm. Despite these drawbacks, this solver already illustrated the capacity of the Eulerian formulation of computing times in shadow zones and retrieving the first arrival in some models where the ray-tracing approach failed.

Several improvements were designed later. van Trier and Symes (1991) proposed an upwind finite-difference scheme based on the observation that the spatial derivatives of the traveltime verify a hyperbolic conservation law. These laws are common in fluid mechanics, and the algorithms to solve them consider the flow direction in the finite-difference approximation: the update of a given point considers only values from upstream in the flow. Intuitively, we may assimilate the flow to the traveltime gradient, and the upwind strategy should be a tool suitable for causality. This results in a more stable algorithm, with no longer need for node ordering on the sides of the expanding box.

In Podvin and Lecomte (1991), another solver that avoids node ordering is presented. A second advantage is that traveltime computation in presence of sharp velocity contrasts is allowed, which was not possible in Vidale's formulas. In this so-called Podvin–Lecomte algorithm, the physical representation of the velocity model is done with square cells of constant slowness. The algorithm is based on an explicit consideration, inside a given cell, of several wavefronts that may occur. This includes transmission, diffraction, and head waves along cell edges. A series of operators is built, one per propagation type. When all the operators are computed, the algorithm picks the one corresponding to the first arrival (lowest traveltime value). In terms of global strategy, the single-pass expanding box formulation is kept, but a reverse propagation step is added in order to take care of turning rays, including recursion if needed. However, this might be inefficient and computationally risky if several levels of recursion were to be involved. Another drawback lies in the list of operators, which do not embed spherical (circular) operators. Therefore, traveltime still exhibits inaccuracies. Despite these limitations, and many other improvements that came later, this algorithm has been widely used among the seismology community: the computer code is provided online (`http://www.geosciences.mines-paristech.fr/fr/recherche/bibliotheque-de-logiciels/geophysique`) and it is easy to handle.

Hole and Zelt (1995) came back to Vidale's initial scheme, and combined it with additional operators from the Podvin–Lecomte algorithm, resulting in an improved accuracy. Moreover, a reverse propagation of traveltime was added after the entire model has been computed, instead of every time a head wave is encountered as done in the Podvin–Lecomte algorithm. Thus, first-arrival times are retrieved in presence of very low-velocity zones, with no consideration of recursion. The resulting method is therefore potentially multi-pass. However, this reverse-propagation procedure must be supervised by the user.

A similar idea was proposed in Kim and Cook (1999), where several ingredients were gathered to obtain a stable algorithm called ENO-DNO-PS. ENO holds for (weighted) essentially non-oscillatory finite-difference schemes developed in Shu and Osher (1988), Shu and Osher (1989), Jiang and Shu (1996), Liu et al. (1994), Jiang and Peng (2000), and suitable for high-order accuracy. This is the first ingredient. The second one, "Down'n'out" (DNO), describes the way the expanding box evolves. Finally, the post-sweeping (PS) treatment is an iterative procedure, hence multi-pass, which consists of updating the traveltime values at nodes in several orderings corresponding to natural orderings of the grid in several plane directions.

From a mathematical point of view, this multi-pass idea with sweeps in alternating directions is at the premise of fast-sweeping method (FSM), relying on global ordering of nodes (Boué and Dupuis, 1999; Tsai et al., 2003; Zhao, 2005; Kao et al., 2004; Luo and Zhao, 2016). Inside each sweep, namely Gauss–Seidel iteration, all nodes are updated following the ordering of the current sweep. The iterative process is known to converge in a finite number of sweeps, although this number is problem-dependent.

Meanwhile, instead of relying on an arbitrary squared-box expansion, wavefront-tracking schemes were proposed to better fit causality by following the expansion of the wavefront itself (Qin et al., 1992; Cao and Greenhalgh, 1994). Doing so, the post-sweeping technique is not necessary. The strategy is single-pass. However, the nodes ordering is essential in order to evolve the front in a causal way: At a given time of the computation, among the nodes that have not been visited yet, the ones to be updated first are the neighbors of the already updated nodes that have the smallest traveltime value. It is worth mentioning here some work based on Dijkstra's algorithm (Dijkstra, 1959) for the traveling salesman problem. Involving graphs theory, some schemes for traveltime computation, like in Moser (1991), do not rely on the Eikonal equation, but on shortest paths in a node-to-node formulation. Combining Dijkstra's-like algorithm with upwind finite-differences for the Eikonal equation led to the fast-marching method (FMM), formally studied in Tsitsiklis (1995) in a Hamilton–Jacobi formulation, and intensively developed in a level-set context in Sethian (1996, 1999), the underlying idea being to track a boundary (the wavefront) evolving in the direction of its normal with a known speed. The FMM idea is the following. The nodes are divided into three sets: accepted nodes, for which the solution has been computed; far nodes which have not been considered yet; and the propagation front area where temporary values at nodes are computed but not yet accepted. The compositions of the three sets evolve at each step of the computation, and steps are performed until all nodes are accepted. Nodes in the front area are ordered with respect to their temporary value. At each step, the node having the lowest value switches to the accepted set, and the values of its neighbors are updated. FMM is therefore a single-pass algorithm, with a crucial ordering of the nodes inside the front area.

State-of-the-art Eikonal solvers now belong to these two families:

- Fast-marching methods (FMM), which are single-pass, but rely on a dynamic ordering of the nodes inside the front. Their complexity is $\mathcal{O}(N \log N)$;
- Fast-sweeping methods (FSM), which do not rely on any dynamic ordering of the nodes, but are multi-pass (iterative). Their complexity is $\mathcal{O}(N)$.

Both FMM and FSM have been intensively applied to solve the Eikonal equation in a wide range of problems. Extensive comparisons showing their numerical efficiencies can be found in Gremaud and Kuster (2006), and highlight that determining which strategy is the best is highly problem-dependent.

Generally, the above-mentioned Eikonal solvers consider computational domains that have rectangular or cuboid shapes. However, the Earth surface is not flat. Even for small scale problems, where the Earth curvature can be reasonably neglected, topography effects may impair inversion results in some regions like mountains, foothills or volcanoes (Improta et al., 2002; Li and Lin, 2016; Trinh et al., 2018). There have been several attempts to develop FD stencils that account for these topography effects (Sethian, 1999; Sun et al., 2011). These developments generally complicate significantly the numerical process and result in loosing the main advantage of FD, namely their simplicity. Instead of designing complex stencils, the topography problem can be solved by changing the equation itself. Lan and Zhang (2013) have proposed to use orthogonal curvilinear coordinates, resulting in a new Eikonal equation which, in the isotropic case, corresponds to a tilted elliptical anisotropic Eikonal equation (see

equation (1.46) later). This can be solved with a Lax–Friedrichs sweeping scheme (Kao et al., 2004), or an upwind scheme based on a Legendre transform (Kao et al., 2005) for more efficiency and accuracy, as performed in Lan and Chen (2017).

The source singularity also needs to be correctly handled. At the source point, the traveltime is equal to zero, and its gradient is not defined: the information flux outgoes from the source to every direction. Therefore, trying to represent such a singularity with standard finite-difference discretization yields a significant inaccuracy. More precisely, high-order numerical schemes always exhibit large errors and are only first-order convergent if no special treatment is applied, due to the singularity of the solution close to the source (see the analysis in Qian and Symes (2002a) for instance). Several studies have been performed in order to mitigate this effect in finite-difference schemes. The celerity transform, initially described by Pica (1997), was then further developed in Zhang et al. (2005a) and promoted under the word factorization in Fomel et al. (2009) and Luo and Qian (2011). Noble et al. (2014) brought together the Podvin–Lecomte algorithm, the fast-sweeping methodology, and the point-source factorization, resulting in an accurate 3D finite-difference solver suitable for media with strong velocity contrasts. In 2.1.2, I give more details regarding the factorization procedure and its numerical implementation.

Applications The first applications of the Eikonal solvers were related to the earthquake location problem. For this problem, the first application of Vidale's algorithm is performed in Nelson and Vidale (1990). The widely used software NonLinLoc (Lomax et al., 2000) is based on the Podvin–Lecomte algorithm.

The Eikonal solvers presented in the previous paragraph have been widely used for tomographic purpose in geophysics (Hole, 1992; Le Meur, 1994; Le Meur et al., 1997; Zelt and Barton, 1998; Leung and Qian, 2006; Taillandier et al., 2009; Lelièvre et al., 2011; Tavakoli F. et al., 2017; Sambolian et al., 2018). In the first implementations, a posteriori ray-tracing is used for the subsequent backprojection of the residuals. Later, the adjoint-state method for the gradient computation in the tomographic inversion (Leung and Qian, 2006; Taillandier et al., 2009) yielded a particularly powerful algorithm when combined with an Eikonal solver, since only one computation is required per source, with no need of ray tracing. The receiver residuals are incorporated into the adjoint computation all at once in the right-hand side.

Many pre-stack depth migration algorithms have also used Eikonal solver, among which Gray and May (1994); Bevc (1997); Buske (1999).

Conclusion The above-mentioned numerical developments are all governed by two main guidelines: computational efficiency and accuracy. The former is obvious in the context of large scale imaging/inversion problems with huge datasets. The latter is required not only for traveltime, which is a smooth quantity, but especially for its gradient, which, again, expresses the direction of propagation of information, and may exhibit severe discontinuities even in simple smooth media. To obtain high accuracy on the traveltime derivatives, one needs at least a second-order scheme. Moreover, amplitude calculation involves the second-order derivatives of the traveltime, as can be seen in the transport equation (1.8). These second-order derivatives relate to the way this flux converges or diverges. Therefore, one should need a third-order Eikonal solver to be able to compute accurate amplitude subsequently. Accuracy is indeed a big challenge. Unfortunately, high-order finite-difference schemes imply a high computational burden. Moreover, they have a wide stencil, which means that a large number of neighbors are used for the update of a single grid point. Hence, in a sense, the solution is smoothed over a

wide area. In order to compute highly-accurate traveltime, we should therefore find alternative numerical techniques to design Eikonal solvers.

1.3 Seismic anisotropy

So far, I have described the theory and algorithms designed in isotropic media, where the wave velocity is a scalar field of the standard space. In this section, I introduce seismic anisotropy, I derive the fundamental equations and I discuss the numerical solutions designed to tackle anisotropy.

1.3.1 Physical description

In an isotropic material, the wave speed is modeled by one parameter for each mode, $V_P(\boldsymbol{x})$ for P waves and $V_S(\boldsymbol{x})$ for S waves. These parameters might vary in space in a inhomogeneous material. However, for various reasons, the propagative materials under study might be more complex, and especially the wave velocity might vary depending on the propagation direction. This phenomenon is called seismic anisotropy, and it is very common in the Earth's interior (Babuska and Cara, 1991). The first mechanism causing anisotropy in the Earth's interior comes from the crystalline structure of minerals (Musgrave, 1970). If the crystals forming a rock are aligned in a preferred orientation, the wave velocity at macroscopic scale will depend on the direction of the wave propagation relatively to the crystal orientation. A second cause of anisotropy results from aligned inclusions of impurities or cracks in rocks (Dey-Barsukov et al., 2000; Schijns et al., 2012). These two phenomena are intrinsic factors of anisotropy. Extrinsic anisotropy, also called apparent anisotropy, is another important physical phenomena, which refers to effects induced on the wave propagation by small heterogeneities. A systematic study was performed by Backus (1962) in presence of thin layering in the medium. When the thickness of each individual layer is small compared to the wavelength, the resulting propagation might be modeled in an effective medium including anisotropy instead of thin layering, with a rotation symmetry around the axis perpendicular to the layering. This way of replacing small-scale model properties by a mean, larger-scale effect as seen by the waves propagating in the medium can be generalized to any kind of heterogeneity, and not only layering. This is referred to as upscaling, and it requires mathematical tools such as the homogenization technique (Capdeville et al., 2010).

1.3.1.1 The general case

In an anisotropic medium, one parameter is no longer enough to describe the velocity of each propagation mode. We need a more complex model for that purpose. As mentioned in section 1.1.1, the isotropic case is a simplification of the general elastic case which involves 21 elastic parameters. This general case allows us to account for anisotropy for triclinic materials, and the parameters of the stiffness tensor write under the form of the elasticity matrix, with the Voigt notation:

$$\boldsymbol{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ & & C_{33} & C_{34} & C_{35} & C_{36} \\ & & & C_{44} & C_{45} & C_{46} \\ & & & & C_{55} & C_{56} \\ & & & & & C_{66} \end{bmatrix}, \tag{1.25}$$

where I do not write the lower diagonal elements since the matrix is symmetric. We want to derive Eikonal and transport equations in such anisotropic media. To do so, we go back to the elastodynamic system (1.1). Following Slawinski (2003), when inserting the ray ansatz into this equation, we obtain again three terms. The first one yields a general Eikonal equation; the second one yields a general transport equation.

Eikonal equation The first Eikonal term writes

$$\sum_{k=1}^{3}\left(\sum_{j=1}^{3}\sum_{l=1}^{3}c_{ijkl}(\boldsymbol{x})T_{,j}T_{,l}-\rho(\boldsymbol{x})\delta_{ik}\right)A_k(\boldsymbol{x})=0, \tag{1.26}$$

with $i\in\{1,2,3\}$, and where $\boldsymbol{A}$ is the displacement vector. Its norm gives the amplitude A, and its direction is along the slowness vector $\boldsymbol{p}=\nabla T$. This system is referred to as Christoffel's equations. Note that the ray ansatz now involves a vector function for amplitude, since we consider the vector displacement field in the elastodynamic system. To get nontrivial solutions, the following relation must be satisfied:

$$\det\left[\sum_{j=1}^{3}\sum_{l=1}^{3}c_{ijkl}(\boldsymbol{x})T_{,j}T_{,l}-\rho(\boldsymbol{x})\delta_{ik}\right]=0, \tag{1.27}$$

for i and k in $\{1,2,3\}$. Slawinski (2003) expresses (1.27) as a factored polynomial in p^2 as

$$\left[p^2-\frac{1}{v_1^2(\boldsymbol{x},\frac{\boldsymbol{p}}{\|\boldsymbol{p}\|})}\right]\left[p^2-\frac{1}{v_2^2(\boldsymbol{x},\frac{\boldsymbol{p}}{\|\boldsymbol{p}\|})}\right]\left[p^2-\frac{1}{v_3^2(\boldsymbol{x},\frac{\boldsymbol{p}}{\|\boldsymbol{p}\|})}\right]=0, \tag{1.28}$$

The three roots $1/v_i^2$ represent the three types of waves that propagate in a general anisotropic medium, with the associated Eikonal equation

$$p^2=\frac{1}{v_i^2(\boldsymbol{x},\frac{\boldsymbol{p}}{\|\boldsymbol{p}\|})}, \tag{1.29}$$

or similarly[2]

$$\|\nabla T\|^2=\frac{1}{v_i^2(\boldsymbol{x},\frac{\nabla T}{\|\nabla T\|})}, \tag{1.30}$$

which is similar to the isotropic Eikonal (1.7), but here the velocity function v_i depends on the direction on propagation described by $\boldsymbol{p}/\|\boldsymbol{p}\|$. The velocity v_i is called the phase velocity since it has the magnitude of the inverse of the magnitude of the slowness vector $\boldsymbol{p}$. The three wave types in an anisotropic medium correspond to one quasi-compressional (qP) mode and two quasi-shear (qS1 and qS2) modes.

In an anisotropic medium, the rays are no longer perpendicular to the wavefronts. In the Hamilton equations (1.19), $\mathrm{d}\boldsymbol{q}/\mathrm{d}t$ is no longer aligned with $\boldsymbol{p}$, because $\partial\mathcal{H}/\partial\boldsymbol{p}$ is not. The energy flux along rays is carried by the group velocity vector $\vec{\mathcal{U}}$, which is parallel to the rays, thus also parallel to $\partial\mathcal{H}/\partial\boldsymbol{p}$. Eikonal equation (1.30) can be reformulated using a Hamiltonian with the same form as (1.21). We get

$$\mathcal{H}(\boldsymbol{x},\nabla T):=\frac{1}{2}\left((\nabla T)^2v_i^2-1\right)=0. \tag{1.31}$$

[2]Please note that in this section, I use indifferently the Lagrangian momentum $\boldsymbol{p}$ or the Eulerian gradient ∇T. However, I use preferentially $\boldsymbol{x}$ as the position in space in the Eulerian viewpoint, instead of $\boldsymbol{q}$ which denotes the position of a Lagrangian particle.

This formulation is particularly interesting since for this particular Hamiltonian we have the equality

$$\vec{\mathcal{U}} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{p}}. \tag{1.32}$$

However, we will see that other formulations of the Hamiltonian might be easier to manipulate. In such cases, a normalization factor appears (see Červený, 2001, eq. 4.7.3), and we obtain

$$\vec{\mathcal{U}} = \frac{\partial \mathcal{H}/\partial \boldsymbol{p}}{\sum_{k=1}^{3} p_k \partial \mathcal{H}/\partial p_k}. \tag{1.33}$$

Transport equation The second term coming from inserting the ray ansatz into the elastodynamic system yields the transport equation which writes

$$\sum_{k=1}^{3}\sum_{j=1}^{3}\sum_{l=1}^{3}\left[\left(c_{ijkl}(\boldsymbol{x})A_l T_{,k}\right)_{,j} + c_{ijkl}(\boldsymbol{x})A_{k,l}T_{,j}\right] = 0. \tag{1.34}$$

Using eigenvectors corresponding to the roots in equation (1.28), we can show that we obtain a transport equation that is analogous to the isotropic equation (1.8), but where ∇T is replaced with the group velocity vector $\vec{\mathcal{U}}$ (Červený, 2001):

$$2\nabla A \cdot \vec{\mathcal{U}} + A\nabla \cdot \vec{\mathcal{U}} = 0. \tag{1.35}$$

Indeed, this equation reflects the conservation of energy, since the group velocity vector carries the energy flux.

We might simplify this general case and look at some common types of symmetries which yield a reduction of the number of parameters. This simplification might be justified by the underlying physical phenomena we want to represent. In the next two sections, I will describe two anisotropic models, and exhibit the resulting Eikonal and transport equations. A systematic analysis of various symmetries yielding other models is presented in Slawinski (2003).

Take-off angle equation In presence of anisotropy, the take-off angle is still constant along a ray, by definition. Thus, the gradient of the take-off angle field is perpendicular to the rays. However, since rays are no longer perpendicular to the wavefront in anisotropic propagation, equation (1.11) is no longer correct. We know from equation (1.33) how to compute the group velocity vector $\vec{\mathcal{U}}$, which governs the energy flux propagation along the rays. Therefore, $\vec{\mathcal{U}}$ is tangent to the rays, so that the gradient of the take-off angle field is perpendicular to $\vec{\mathcal{U}}$. The general take-off angle equation then writes

$$\nabla_{\boldsymbol{x}} \phi_{\boldsymbol{s}}(\boldsymbol{x}) \cdot \vec{\mathcal{U}}(\boldsymbol{x}) = 0. \tag{1.36}$$

Again, this is an HJ equation of the form of equation (1.9).

1.3.1.2 Transversely isotropic medium

The vertical case A transversely isotropic (TI) medium is invariant with respect to a rotation around one axis. In the plane normal to this axis, the medium is isotropic. I first consider the vertical axis as the rotation axis. This case is called vertical transversely isotropic (VTI). This case is appropriate

for the description of a horizontal thin layering in terms of apparent anisotropy (Backus, 1962). When considering this type of symmetry, the resulting elasticity matrix contains only five independent elastic parameters and writes

$$\boldsymbol{C}_{VTI} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{13} & 0 & 0 & 0 \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{C_{11}-C_{12}}{2} \end{bmatrix}. \tag{1.37}$$

Consider an arbitrary direction of propagation given by the unit vector $\boldsymbol{n} = (n_x, n_y, n_z)$ such that $\boldsymbol{n} = \boldsymbol{p}/\|\boldsymbol{p}\|$. Christoffel's equation (1.27) then writes

$$\begin{aligned} &\Big(C_{66}(n_x^2+n_y^2) + C_{44}n_z^2 - \rho v^2\Big) \\ &\Big(- C_{13}^2(n_x^2+n_y^2)n_z^2 - 2C_{13}C_{44}(n_x^2+n_y^2)n_z^2 \\ &\quad + C_{33}C_{44}n_z^4 - C_{44}(n_x^2+n_y^2)\rho v^2 - C_{33}n_z^2\rho v^2 - C_{44}n_z^2\rho v^2 \\ &\quad + C_{11}(n_x^2+n_y^2)\big(C_{44}(n_x^2+n_y^2) + C_{33}n_z^2 - \rho v^2\big) + \rho^2 v^4\Big) = 0. \end{aligned} \tag{1.38}$$

The rotational symmetry means that we can select any plane containing the vertical symmetry axis for propagation, with no loss of generality. Now assuming a propagation inside the xz-plane, we let $n_y = 0$. Equation (1.38) exhibits two factors. The first one corresponds to SH propagation (shear wave propagating inside the xz-plane with a purely orthogonal displacement direction parallel to the y-axis). The second one couples both qP (quasi-pure compressional wave with a displacement direction contained in the xz-plane and quasi-parallel to the direction of propagation) and qSV (quasi-pure shear wave with a displacement direction contained in the xz-plane and quasi-orthogonal to the direction of propagation) modes. These two factors can be solved independently. In view of the computation of first-arrival traveltime, I consider here the coupled P–SV mode which contains the compressional mode, and I define

$$\begin{aligned} &V_P = \sqrt{\frac{C_{33}}{\rho}}, \quad V_S = V_{SV} = \sqrt{\frac{C_{44}}{\rho}}, \\ &\epsilon = \frac{C_{11}-C_{33}}{2C_{33}}, \quad \delta = \frac{(C_{13}+C_{44})^2 - (C_{33}-C_{44})^2}{2C_{33}(C_{33}-C_{44})}. \end{aligned} \tag{1.39}$$

The quantities V_P and V_S denote the P- and SV-wave velocities along the rotation-symmetry axis, while ϵ and δ are known as the Thomsen's parameters (Thomsen, 1986). The P–SV Eikonal for a propagation in the xz-plane in a VTI medium can be finally written as

$$aT_{,x}^4 + bT_{,z}^4 + cT_{,x}^2T_{,z}^2 + dT_{,x}^2 + eT_{,z}^2 - 1 = 0, \tag{1.40}$$

where

$$\begin{cases} a = -(1+2\epsilon)V_P^2V_S^2, \\ b = -V_P^2V_S^2, \\ c = -(1+2\epsilon)V_P^4 - V_S^4 + (V_P^2 - V_S^2)\left[V_P^2(1+2\delta) - V_S^2\right], \\ d = V_S^2 + (1+2\epsilon)V_P^2, \\ e = V_P^2 + V_S^2. \end{cases} \tag{1.41}$$

Equivalent formulations might be found in Postma (1955), Musgrave (1970), Payton (1983), Carcione et al. (1988), Červený (2001), and Slawinski (2003). For a formulation in the 3D space, $T_{,x}^2$ has to be replaced by $T_{,x}^2 + T_{,z}^2$. I keep hereafter the 2D formulation for the sake of simplicity.

Setting $\epsilon = \delta = 0$, we retrieve the isotropic case with two roots corresponding to P waves and S waves. When $\epsilon = \delta \neq 0$, the coefficient c in front of the cross-term $T_{,x}^2 T_{,z}^2$ cancels out, so that we obtain the so-called elliptical anisotropy. This particular case is equivalent to a simple dilation applied to the isotropic case along the axis orthogonal to the rotation-symmetry axis. This case is of little physical interest as discussed by Levin (1979). In a typical VTI medium, we have $\epsilon > \delta$, and corresponding anelliptic effects are important to account for.

As considered by Alkhalifah (2000), in view of first-arrival traveltime computation, equation (1.40) can be simplified for the acoustic case, and this does not perturb the numerical solution for the P-wave propagation. Since the compressional velocity is always higher than the shear velocity, the first-arrival traveltime is not affected. Hence, for the acoustic case, we can set $V_S = 0$, and we obtain

$$\begin{cases} a = 0, \\ b = 0, \\ c = -2(\epsilon - \delta) V_P^4, \\ d = (1 + 2\epsilon) V_P^2, \\ e = V_P^2. \end{cases} \tag{1.42}$$

The VTI Eikonal becomes

$$d T_{,x}^2 + e T_{,z}^2 + c T_{,x}^2 T_{,z}^2 - 1 = 0. \tag{1.43}$$

I show some VTI wavefronts in a homogeneous medium for various sets of Thomsen's parameters in figure 1.8. These plots are computed using the parametric formulation in Carcione et al. (1988, eq. 5.9). A change in ϵ directly affects the horizontal velocity through the parameter d in equation (1.43). This is visible in figure 1.8a. A change in δ does not affect the horizontal nor vertical velocities but the shape of the wavefront in the diagonal directions, through the anellipticity related to $\epsilon - \delta$. This is visible in figure 1.8b.

Group velocity vector In the Hamiltonian formalism, equation (1.43) is a Hamilton–Jacobi equation with the Hamiltonian

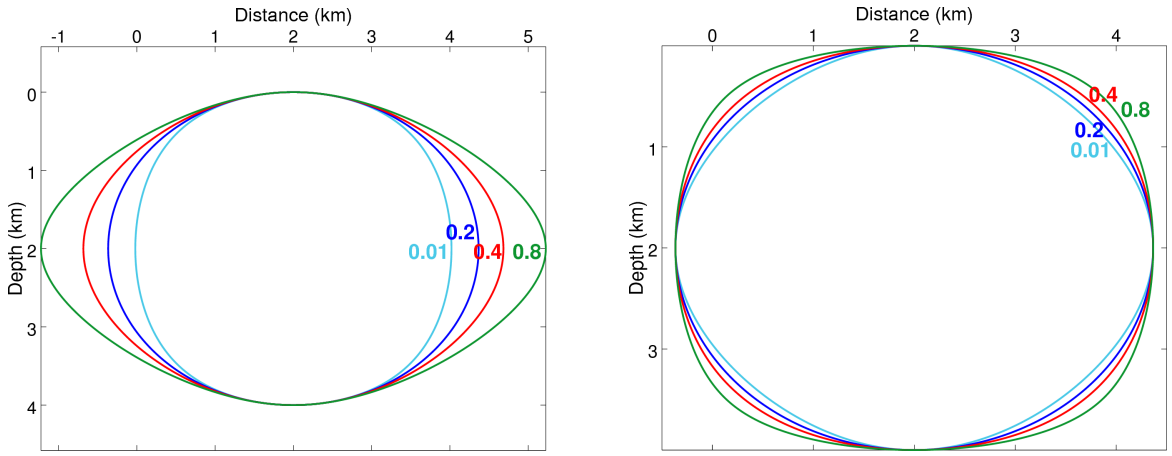
$$\mathcal{H}(\boldsymbol{x}, \nabla T) = d T_{,x}^2 + e T_{,z}^2 + c T_{,x}^2 T_{,z}^2 - 1. \tag{1.44}$$

In the 2D VTI case, in view of equation (1.33), we may write the group velocity vector

$$\vec{\mathcal{U}} = \frac{1}{d T_{,x}^2 + e T_{,z}^2 + 2c T_{,x}^2 T_{,z}^2} \begin{pmatrix} d T_{,x} + c T_{,x} T_{,z}^2 \\ e T_{,z} + c T_{,x}^2 T_{,z} \end{pmatrix}. \tag{1.45}$$

For the VTI case, we may now insert this expression into equation (1.35) to obtain a transport equation, and into equation (1.36) to obtain a take-off angle equation.

The tilted case Due to the gravity of the Earth, sedimentary layers are naturally horizontally deposited. However, tectonic forces, among other processes occurring inside the crust, may cause tilting and rotation of these layers. This results in a TI medium with a tilted axis, namely a tilted transversely isotropic (TTI) medium. To represent this tilt, we introduce a local tilt angle $\theta(\boldsymbol{x})$ as an additional



(a) Wavefront curves for $\delta = 0.2$ and various values of ϵ.

(b) Wavefront curves for $\epsilon = 0.2$ and various values of δ.

Figure 1.8: Wavefront curves in homogeneous VTI media. When $\epsilon = \delta$ the solution is an ellipse. Recall that in the Earth, $\epsilon > \delta$.

parameter to the previous model. We then apply a rotation operator to the traveltime derivatives in equation (1.43) and obtain the TTI Eikonal equation

$$\begin{aligned} &d(T_{,x}\cos\theta + T_{,z}\sin\theta)^2 + e(T_{,z}\cos\theta - T_{,x}\sin\theta)^2 \\ &+c(T_{,x}\cos\theta + T_{,z}\sin\theta)^2(T_{,z}\cos\theta - T_{,x}\sin\theta)^2 - 1 = 0. \end{aligned} \tag{1.46}$$

The tilted transport and take-off angle equations can be derived as well by applying the same rotation.

1.3.1.3 Orthorhombic medium

The vertical case 3D orthorhombic media have three mutually orthogonal planes of mirror symmetry. Nine independent elastic parameters are involved int the elasticity matrix which writes (Musgrave, 1970):

$$\boldsymbol{C}_{VTI} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{bmatrix}. \tag{1.47}$$

This description is appropriate for the description of one or several systems of cracks. The description of fractured media using apparent orthorhombic anisotropy was known from theoretical studies of Kachanov (1980) and confirmed numerically by Grechka et al. (2006). In exploration geophysics, orthorhombic anisotropy is of high interest since many sedimentary formations like sands and carbonates contain fracture sets, and fractured reservoirs would contain one-third of the world's hydrocarbon reserves (Bakulin et al., 2000; Tsvankin et al., 2010). Geomechanical models of cracked/fractured media might involve stress-induced orthorhombic anisotropy (Thanoon et al., 2015).

A study of qP-wave propagation in orthorhombic media was performed in Tsvankin (1997). An acoustic derivation for qP waves was proposed in Alkhalifah (2003) by setting the qS-wave velocity to

zero along the symmetry axis with no effect on the accuracy of the qP-wave traveltime computation. Following Waheed et al. (2015b) but using a different parameterization, we may write the vertical orthorhombic (VOR) Eikonal as

$$\begin{aligned} a(T_{,x})^2 + b(T_{,y})^2 + c(T_{,z})^2 + d(T_{,x})^2(T_{,y})^2 + e(T_{,x})^2(T_{,z})^2 \\ + f(T_{,y})^2(T_{,z})^2 + g(T_{,x})^2(T_{,y})^2(T_{,z})^2 - 1 = 0, \end{aligned} \tag{1.48}$$

with the following parameters:

$$\begin{cases} a = V_P^2(1+2\epsilon_2), \\ b = V_P^2(1+2\epsilon_1), \\ c = V_P^2, \\ d = V_P^4(1+2\epsilon_2)\big((1+2\epsilon_2)(1+2\delta) - (1+2\epsilon_1)\big), \\ e = -2(\epsilon_2 - \delta_2)V_P^4, \\ f = -2(\epsilon_1 - \delta_1)V_P^4, \\ g = -V_P^6\big((1+2\epsilon_2)^2(1+2\delta) - 2(1+2\epsilon_2)\sqrt{(1+2\delta_2)}\sqrt{(1+2\delta_1)}\sqrt{(1+2\delta)} \\ \quad + (1+2\delta_2)(1+2\delta_1) - 4(\epsilon_2 - \delta_2)(\epsilon_1 - \delta_1)\big), \end{cases} \tag{1.49}$$

where anisotropic parameters ϵ_2 and δ_2 hold in the $[x, z]$ plane, anisotropic parameters ϵ_1 and δ_1 hold in the $[y, z]$ plane, and anisotropic parameter δ holds in the $[x, y]$ plane. This parameterization comes from Tsvankin (1997) and is consistent with an alternative parameterization proposed in Alkhalifah (2003) and Waheed et al. (2015b).

Group velocity vector In the Hamiltonian formalism, equation (1.48) is a Hamilton–Jacobi equation with the Hamiltonian

$$\begin{aligned} \mathcal{H}(\boldsymbol{x}, \nabla T) = & a(T_{,x})^2 + b(T_{,y})^2 + c(T_{,z})^2 + d(T_{,x})^2(T_{,y})^2 + e(T_{,x})^2(T_{,z})^2 \\ & + f(T_{,y})^2(T_{,z})^2 + g(T_{,x})^2(T_{,y})^2(T_{,z})^2 - 1. \end{aligned} \tag{1.50}$$

In view of equation (1.33), the group velocity vector then writes

$$\vec{\mathcal{U}} = \frac{1}{A(\boldsymbol{x}, \nabla T)} \begin{pmatrix} aT_{,x} + dT_{,x}T_{,y}^2 + eT_{,x}T_{,z}^2 + gT_{,x}T_{,y}^2T_{,z}^2 \\ bT_{,y} + dT_{,x}^2T_{,y} + fT_{,y}T_{,z}^2 + gT_{,x}^2T_{,y}T_{,z}^2 \\ cT_{,z} + eT_{,x}^2T_{,z} + fT_{,y}^2T_{,z} + gT_{,x}^2T_{,y}^2T_{,z} \end{pmatrix}, \tag{1.51}$$

with

$$\begin{aligned} A(\boldsymbol{x}, \nabla T) = & \, a(T_{,x})^2 + b(T_{,y})^2 + c(T_{,z})^2 + 2d(T_{,x})^2(T_{,y})^2 + 2e(T_{,x})^2(T_{,z})^2 \\ & + 2f(T_{,y})^2(T_{,z})^2 + 3g(T_{,x})^2(T_{,y})^2(T_{,z})^2. \end{aligned} \tag{1.52}$$

For the VOR case, we can insert this expression into equation (1.35) to obtain a transport equation, and into equation (1.36) to obtain a take-off angle equation.

The tilted case The tilted (TOR) case is retrieved by applying the local 3×3 rotation operator involving three rotation angles: the dip angle $\theta(\boldsymbol{x})$, the azimuth angle $\phi(\boldsymbol{x})$, and the rotation angle $\psi(\boldsymbol{x})$

which is the rotation angle of the elastic tensor in the rotated horizontal plane, corresponding to the crack orientation in this plane. The rotation operator $\mathcal{R}_{\text{TOR}}$ writes

$$\mathcal{R}_{\text{TOR}} = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.53}$$

Replacing $u_{,x}$, $u_{,y}$, and $u_{,z}$ by their rotated expression from (1.53) into OR Eikonal equation (1.48) yields the TOR Eikonal equation. The substitution is as simple as for going from (1.43) to (1.46), although I do not explicitly write the TOR Hamiltonian here for the sake of concision.

Again, the tilted transport and take-off angle equations can be derived as well by applying the same rotation.

1.3.2 Practical implications and numerical developments

In an anisotropic configuration, the energy flux is no longer perpendicular, in the Euclidean metric, to the wavefront as in the isotropic case. Therefore, the front propagation is characterized by two vectors. The first one is the phase slowness vector $\boldsymbol{p} = \nabla T$, oriented towards the normal of the wavefront. The second one is the group velocity vector $\vec{U}$, which carries the high-frequency part of the energy flux, and computed from equation (1.33).

Regarding numerical algorithms, extensions to the anisotropic configuration have been designed for the Lagrangian, semi-Lagrangian and Eulerian approaches described in section 1.2. Anisotropic ray tracing theory requires the general Hamiltonian formalism. A perturbation approach to elliptical anisotropy was proposed by Farra (1989), and by Mensch et al. (1998) for 3D orthorhombic media. Mensch and Farra (2002) employed this perturbation approach and perform P-wave tomography in weakly orthorhombic media. Gajewski and Pšenčík (1990) solved the exact dynamic ray tracing system for inhomogeneous anisotropic media for the computation of vertical seismic profiles in 3D laterally varying structures. More recently, depth imaging in orthorhombic media has gained interest in the exploration geophysics community regarding fractured reservoirs. For instance, orthorhombic kinematic ray tracing has been used in a tomographic application by Li et al. (2012) for velocity model building, applying a tilted orthorhombic tomography method to a real dataset from a fractured region in the northwestern gulf of Mexico.

The wavefront tracking method based on Huygens's principle proposed in Qin and Schuster (1993) handles anisotropy. In the propagation step, the group velocity direction must be found from the orientation of each elementary segment in order to propagate the wavefront in that direction.

In presence of anisotropy, the Eikonal equation is more complex and needs adequate numerical strategies. The ENO-DNO-PS principles were extended to the anisotropic case in Dellinger and Symes (1997), Kim (1999), and Qian et al. (2001). Some extensions have also been carried out regarding FMM (Cristiani, 2009), however they are based on approximations and they are generally difficult to implement. Mirebeau and Portegies (2018) proposed a new approach for the FMM for anisotropic Eikonal. However, the most mature strategies proposed so far in the geophysical context for anisotropy rely on FSM. The first extensions consider elliptical anisotropy (Tsai et al., 2003; Qian et al., 2007a), which could be handled quite naturally since it amounts to a dilation in space. Hence, the design of a local solver relies on finding the roots of quadratic equations. However, the more general cases of anelliptical TTI or TOR Eikonal comprise spatial derivatives of the traveltime respectively to the power of four or six, which is more challenging.

Several approaches have been proposed, either by solving the quartic equations and selecting the appropriate roots (Han et al., 2017), yielding a high computational load, by treating the anellipticity as a perturbation to the elliptical case and solving for the corresponding time expansion (Waheed et al., 2015a), or by implementing a fixed-point iteration technique which solves an elliptical equation at each iteration with an appropriate right-hand side accounting for anellipticity (Tavakoli F. et al., 2015; Waheed et al., 2015b; Waheed and Alkhalifah, 2017). Hu et al. (2017) promoted another approach based on dynamic programming for traveltime computation in orthorhombic media which is suitable for depth migration since it computes the traveltime from one depth to the next depth. However, the first arrival is not always guaranteed, depending on which strategy is employed.

1.4 Conclusion

Among the equations I have presented in the high-frequency approximation, the non-linear Eikonal one is the most challenging to solve. Once done, the computation of amplitude and/or take-off angle is easier, due to the linearity of the related equations.

The robustness of Eulerian methods for first-arrival traveltime computation has made them the methods of choice for traveltime computation in several applications such as adjoint traveltime/slope tomography and Kirchhoff migration, for imaging and inversion in seismology and velocity model building prior to full-waveform inversion or reverse-time migration at exploration scales.

In complex media, the computation is still very challenging when considering non flat topographies and anisotropic media. Compromises have to be found between implementation complexity, computational efficiency, and accuracy. Therefore, it is desirable to consider alternative numerical techniques to design Eikonal solvers.

In the two following chapters of this thesis, I propose to go beyond the current limitations of Eikonal solvers by introducing a new numerical framework using a Hamiltonian formulation which is able to handle the most general cases of complexity, including anisotropy and complex topographies, while achieving an unprecedented accuracy, and with a very good efficiency.

Chapter 2

Solving 2D Eikonal equations with a discontinuous Galerkin method for highly accurate computation of traveltime

Contents

Introduction

In this chapter, I describe a high-order numerical scheme proposed in the applied mathematics community to solve Hamilton–Jacobi partial differential equations. Since the Eikonal equations presented in chapter 1 belong to this family, I use this scheme and adapt it to the 2D geophysical problem. The materials presented in this chapter are largely extracted from two first publications on this topic, Le Bouteiller et al. (2017, 2018a). At the beginning of this work, I was provided with a prototype code from Mondher Benjemaa (University of Sfax, Tunisia), to whom I am very grateful for his highly valuable assistance.

Notations In this chapter, I introduce a general Hamiltonian formalism in a pseudo-time-dependent formulation. The generic Hamilton–Jacobi equations writes

$$\partial_\xi u(\boldsymbol{x},\xi) + \mathcal{H}(\boldsymbol{x}, \nabla_{\boldsymbol{x}} u(\boldsymbol{x},\xi)) = 0, \tag{2.1}$$

where ξ is the pseudo-time variable, $u(\boldsymbol{x},\xi)$ is the unknown function of time and space, and $\mathcal{H}(\boldsymbol{x}, \nabla_{\boldsymbol{x}} u(\boldsymbol{x},\xi))$ is the Hamiltonian. For traveltime computation, we look for stationary states of equation (2.1) with suitable Hamiltonian detailed in the next section. At the stationary state, we may write

$$\lim_{\xi\to\infty} u(\boldsymbol{x},\xi) \equiv T(\boldsymbol{x}), \tag{2.2}$$

where $T(\boldsymbol{x})$ is the traveltime field. In a finite computational domain, we set the source boundary condition to $u(\boldsymbol{x}_s,\xi) = 0$ at any pseudo-time ξ at the source point $\boldsymbol{x}_s$. Such stationary state is obtained at a finite pseudo-time $\xi^\star$ once the source information has been propagated from the source to the entire domain. Using this time-marching procedure to reach the steady-state is studied in Zhang et al. (2005c), while a formal link between the static and the dynamic Hamilton–Jacobi equations is proposed by Osher (1993) through the level-set framework.

State of the art Reaching high-order accuracy when solving Eikonal equation with finite-difference methods is challenging, partly due to the non-compactness of the resulting stencils, and the lack of flexibility of such method regarding unstructured meshes and complex geometries. This is why other types of numerical methods are considered. They involve a weak formulation of the equation. Among these methods, finite-element methods (FEM) allow to reach high-order accuracy. However, the continuous formulation of FEM exhibits some drawbacks such as the fact that it gives rise to a system of equations that have to be solved implicitly. In presence of discontinuities of the spatial gradient of traveltime in the Eikonal solution, solving this system is inefficient due to the strong underlying non-linearity. For these reasons, I focus on discontinuous Galerkin methods, which are able to accurately capture highly complex solutions with discontinuities. A clear tutorial on these methods is proposed by Cockburn (2003). In particular, the Runge–Kutta discontinuous Galerkin (RK-DG) method is proven to be very appropriate for time-dependent HJ equations (2.1) (see the review by Cockburn and Shu, 2001). The RK-DG method relies on the method of lines (Schiesser, 1991), where the problem is firstly discretized in space, yielding a system of ordinary differential equations (ODE) in time; this system is then solved in an explicit way with a Runge–Kutta solver.

Initially developed for solving hyperbolic conservation laws (Cockburn and Shu, 1998), the RK-DG method was then applied to solve the conservation law verified by the derivatives of the solution of the HJ equation (2.1), in order to reconstruct the solution itself afterward (Hu and Shu, 1999). Efforts were

made by Cheng and Shu (2007) to recover directly the solution of equation (2.1) in order to simplify the scheme. However, the method would suffer from an entropy violation issue in some cells, which had to be corrected by a specific *ad-hoc* procedure. Cheng and Wang (2014) achieved a new step for directly solving equation (2.1) for the needed quantity $u(\boldsymbol{x}, \xi)$ with a DG method. An entropy fix is embedded inside the weak formulation itself, which greatly simplifies the implementation with a compact scheme. For these reasons, I consider the method proposed by Cheng and Wang (2014) in my work.

Outline The outline of this chapter is the following:

- In section 2.1, I give the Eikonal equations in their time-dependent Hamilton–Jacobi formulation, and I describe how I apply the factorization technique to these equations, which is essential for accuracy.
- In section 2.2, I describe the RK-DG method coming from Cheng and Wang (2014), and I give details on the improvements I bring to the original scheme in order to make it suitable to the geophysical problem of traveltime computation. These improvements are related to the choice of the Hamiltonian in order to optimize the computation efficiency, the inclusion of suitable boundary conditions through an additional term in the original scheme, and the practical implementation of the point-source condition and the initial condition.
- In section 2.3, I show the results of an extensive convergence analysis which validates the accuracy of the RK-DG scheme, for various source treatments, Hamiltonians, polynomial orders, and mesh characteristics.
- In section 2.4, I illustrate the good behavior of the RK-DG solver with various examples in complex anisotropic media.
- Finally, in section 2.5, I conclude on the RK-DG method and give perspectives related to its poor computational efficiency.

2.1 Factored Eikonal equations for 2D isotropic and TTI media

2.1.1 Hamilton–Jacobi equations

Isotropic case In an isotropic medium with a wave speed $c(\boldsymbol{x})$, the Hamiltonian can be written as

$$\mathcal{H}(\boldsymbol{x}, \nabla_{\boldsymbol{x}} u) = \|\nabla_{\boldsymbol{x}} u\| - \frac{1}{c(\boldsymbol{x})}, \tag{2.3}$$

and the corresponding time-dependent Hamilton–Jacobi equation writes

$$\partial_\xi u + \|\nabla_{\boldsymbol{x}} u\| - \frac{1}{c(\boldsymbol{x})} = 0. \tag{2.4}$$

In view of equation (1.7), the stationary state of (2.4) verifies Eikonal equation $\mathcal{H} = 0$. Since we are only interested in the stationary state of (2.1), we are free to consider other Hamiltonians which yield the same final state, such as

$$\mathcal{H}(\mathbf{x}, \nabla_{\boldsymbol{x}} u) = c(\boldsymbol{x}) \, \|\nabla_{\boldsymbol{x}} u\| - 1, \tag{2.5}$$

and the corresponding time-dependent Hamilton–Jacobi equation

$$\partial_\xi u + c(\boldsymbol{x}) \left\| \nabla_{\boldsymbol{x}} u \right\| - 1 = 0. \tag{2.6}$$

Equation (2.6) describes the propagation of a front with the local speed $v(\boldsymbol{x}) = c(\boldsymbol{x})$, whereas in (2.4) the front propagates with a uniform speed $v = 1$. This might impact upon the computational efficiency, although the steady-state solution would be the same in both cases. The best choice is discussed further in section 2.2.2.

2D TTI case Equation (1.46) is the 2D Eikonal equation for TTI media. The corresponding Hamiltonian writes

$$\begin{aligned} \mathcal{H}_{TTI}(\mathbf{x}, \nabla_{\boldsymbol{x}} u) =& d(u_{,x}\cos\theta + u_{,z}\sin\theta)^2 + e(u_{,z}\cos\theta - u_{,x}\sin\theta)^2 \\ &+ c(u_{,x}\cos\theta + u_{,z}\sin\theta)^2(u_{,z}\cos\theta - u_{,x}\sin\theta)^2 - 1. \end{aligned} \tag{2.7}$$

2.1.2 Point-source factorization

2.1.2.1 Factored Eikonal equations

As explained in section 1.2.3, the gradient of the traveltime is not defined at the source point. The corresponding singularity is known to impair accuracy and convergence orders of numerical solutions of Eikonal equation. To retrieve expected convergence orders of a numerical scheme, one might initialize the traveltime solution analytically near the source inside a grid-independent area. This is the method promoted for numerical validation in, e.g., Vidale (1988b), van Trier and Symes (1991), Qian and Symes (2002b), Kao et al. (2004), Zhao (2005), and Zhang et al. (2006), as well as in the convergence analysis presented in section 2.3.1. However, this relies on an *ad-hoc* radius parameter around the source with no guidances for setting its value, and in practice the velocity might not be constant in such an area. Adaptive grid refinement method was proposed in Qian and Symes (2002a). However, the simplest and most efficient way to tackle the point-source singularity is the factorization technique. This technique yields no extra computational load, no *ad-hoc* parameter, no assumption related to the velocity model around the source, and provides a remarkable gain in terms of accuracy.

Next, I describe the factorization technique and I extend the work of Pica (1997), Zhang et al. (2005a), Fomel et al. (2009), and Luo and Qian (2011) to the discontinuous Galerkin framework for the time-dependent Hamilton–Jacobi equation. The factorization principle is to decompose the traveltime solution into two functions: a reference solution which should embed the point-source singularity, and the numerical solution, sometimes called perturbation[1].We choose a reference function $u_0(\boldsymbol{x})$, which does not depend on pseudo-time ξ. Its purpose is to describe the traveltime solution in the vicinity of the source in an accurate way. The traveltime solution is then computed as a sum (additive factorization), or a multiplication (multiplicative factorization), of the reference solution $u_0(\boldsymbol{x})$ and the numerical solution $\tau(\boldsymbol{x}, \xi)$ which is the unknown of the numerical scheme. The additive decomposition, proposed in Luo and Qian (2012), writes

$$u(\boldsymbol{x}, \xi) = u_0(\boldsymbol{x}) + \tau(\boldsymbol{x}, \xi), \tag{2.8}$$

while the multiplicative decomposition writes

$$u(\boldsymbol{x}, \xi) = u_0(\boldsymbol{x})\tau(\boldsymbol{x}, \xi). \tag{2.9}$$

[1]Far from the source, this "perturbation" might be significant in an arbitrary medium!

Plugging expression (2.9) or (2.8) into time-dependent Hamilton–Jacobi equation (2.1) yields a new equation to be solved for the solution field $\tau(\boldsymbol{x}, \xi)$. Combining the additive formulation (2.8) and the isotropic time-dependent Hamilton–Jacobi equation (2.4) yields

$$\partial_\xi \tau + \|\nabla_{\boldsymbol{x}} u_0 + \nabla_{\boldsymbol{x}} \tau\| - \frac{1}{c(\boldsymbol{x})} = 0, \tag{2.10}$$

with the boundary and initial conditions

$$\begin{aligned} \tau(\boldsymbol{x_s}, \xi) &= 0, \\ \tau(\boldsymbol{x}, 0) &= 0, \end{aligned} \tag{2.11}$$

where $\boldsymbol{x_s}$ denotes the point source location. The new equation (2.10) for $\tau(\boldsymbol{x}, \xi)$ is of the same Hamilton–Jacobi type as the original one (2.4) for $u(\boldsymbol{x}, \xi)$. Using the multiplicative formulation (2.9) yields

$$\partial_\xi \tau + \|u_0 \nabla_{\boldsymbol{x}} \tau + \tau \nabla_{\boldsymbol{x}} u_0\| - \frac{1}{c(\boldsymbol{x})} = 0. \tag{2.12}$$

Equation (2.12) contains additional complexities out of the frame of equation (2.1). This comes from the expression $\nabla(u_0 \tau)$ which cannot be written in terms of derivatives of τ, but also involves τ itself. For this reason, the additive formulation (2.8) is preferred in what follows.

The additive strategy applied to TTI Hamiltonian of equation (2.7) leads to

$$\begin{aligned} \partial_\xi \tau + d[(u_{0,x} + \tau_{,x}) \cos\theta + (u_{0,z} + \tau_{,z}) \sin\theta]^2 + e[(u_{0,z} + \tau_{,z}) \cos\theta - (u_{0,x} + \tau_{,x}) \sin\theta]^2 \\ + c[(u_{0,x} + \tau_{,x}) \cos\theta + (u_{0,z} + \tau_{,z}) \sin\theta]^2 [(u_{0,z} + \tau_{,z}) \cos\theta - (u_{0,x} + \tau_{,x}) \sin\theta]^2 - 1 = 0. \end{aligned} \tag{2.13}$$

If the reference solution is appropriate, then the numerical solution $\tau(\boldsymbol{x}, t)$ should be "less" singular at the source. In other words, the pollution induced by the point-source condition in all the medium should significantly decrease, yielding a gain in convergence order. This is clearly illustrated in the convergence analysis of section 2.3.1.

2.1.2.2 Choice of the reference solution

Isotropic case In an arbitrarily heterogeneous isotropic medium, intuitively, a low-order approximation of the solution around the source should be obtained by considering a locally homogeneous medium in a small vicinity of the source. This is a reasonable approximation in the high-frequency regime where the velocity model should be smooth. Therefore, the reference solution $u_0(\boldsymbol{x})$ to be used in equation (2.10) would be the distance function to the source point $\boldsymbol{x}_s$ divided by the velocity at the source:

$$u_{0_{\text{ISO}}}(\boldsymbol{x}) = \frac{\text{dist}(\boldsymbol{x}, \boldsymbol{x}_s)}{c(\boldsymbol{x})}. \tag{2.14}$$

This is the reference function used in Fomel et al. (2009). More formally, Luo et al. (2014b) assume power series expansions of traveltime squared $\mathcal{T}$ and slowness squared $\mathcal{S}$ around the source as

$$\begin{aligned} \mathcal{T}(\boldsymbol{x}) &= \sum_{\nu=0}^{\infty} \mathcal{T}_\nu(\boldsymbol{x}), \\ \mathcal{S}(\boldsymbol{x}) &= \sum_{\nu=0}^{\infty} \mathcal{S}_\nu(\boldsymbol{x}), \end{aligned} \tag{2.15}$$

where $\mathcal{T}_\nu(\boldsymbol{x})$ and $\mathcal{S}_\nu(\boldsymbol{x})$ are homogeneous polynomials of degree ν in $\boldsymbol{x}$. The static Eikonal equation writes

$$\mathcal{S}\mathcal{T} = \frac{1}{4}|\nabla\mathcal{T}|^2. \tag{2.16}$$

Replacing $\mathcal{T}$ and $\mathcal{S}$ in equation (2.16) by their polynomial expansions (2.15) and equating the terms of equal degree, they show that:

1. the distance function (2.14) commonly used as the reference solution in the factorization procedure is the zeroth-order approximation in this formulation;

2. it is possible to increase the order by building analytically other reference solutions which take into account higher-order terms of the slowness squared expansion.

The first point comes from successive results $\mathcal{T}_0 = 0$, $\mathcal{T}_1 = 0$, and $\mathcal{T}_2 = \mathcal{S}_0\,\boldsymbol{x}^2$ which corresponds to the distance function (2.14). The second point comes from a recursive relationship between higher-order terms.

Most of the time, using a distance function will be enough for our purposes. However, I show in the convergence analysis of section 2.3.2 that it is possible in practice to increase the order of accuracy of the solution when using a higher-order reference solution.

TI case For the anisotropic case, Luo and Qian (2012) considered a factored anisotropic Eikonal equation, but only for elliptical media (where $\epsilon = \delta$). Following their approach, the reference solution $u_0(\boldsymbol{x})$ in equation (2.13) might be chosen as the exact solution in a homogeneous elliptical TI medium of velocity $V_P(\boldsymbol{x}_s)$, thus accounting for $V_P(\boldsymbol{x}_s)$, $\epsilon(\boldsymbol{x}_s)$, and $\theta(\boldsymbol{x}_s)$, but not $\delta(\boldsymbol{x}_s)$. The VTI case, for $\theta(\boldsymbol{x}_s) = 0$, writes

$$u_{0_{\text{VTI}}}(\boldsymbol{x}) = \sqrt{\frac{(x-x_s)^2}{V_P(\boldsymbol{x}_s)\sqrt{1+2\epsilon}} + \frac{(z-z_s)^2}{V_P(\boldsymbol{x}_s)}}. \tag{2.17}$$

The tilted case is retrieved by the local rotation of coordinates

$$\begin{aligned} x - x_s &\to (x-x_s)\cos\theta(\mathbf{x}_s) + (z-z_s)\sin\theta(\mathbf{x}_s), \\ z - z_s &\to (z-z_s)\cos\theta(\mathbf{x}_s) - (x-x_s)\sin\theta(\mathbf{x}_s). \end{aligned} \tag{2.18}$$

The efficiency of this factorization for the TTI case is illustrated in section 2.4.1.

If the anisotropy is elliptical around the source, the reference solution (2.17) would well capture the source singularity. If not, a source effect due to anellipticity would remain. In order to increase the accuracy in presence of anellipticity (which is always the case in geophysical TI media, see Levin (1979) and section 1.3.1.2), we can also use exact solution of the anelliptical equation in a homogeneous medium as the reference solution $u_0(\boldsymbol{x})$. However, this solution is not known in closed form but only parametrically (see e.g. Carcione et al., 1988). In section 2.4.1, I illustrate how to pre-compute the anelliptical reference solution at all discrete points where values are needed, which yields a significant improvement of the results accuracy.

2.2 A discontinuous Galerkin scheme for Hamilton–Jacobi equations

2.2.1 The original scheme

Space discretization In the following, I describe the formulation of the scheme for a 2D unstructured triangular mesh, and I detail the meaning of the contributions of each term of the scheme. In the general case, the two-dimensional spatial domain Ω is partitioned into n triangular elements (also called cells), denoted by $K_i, i = 1, ..., n$. A local approximation space $\mathcal{P}_i$ of dimension d_i is chosen for each element K_i together with a basis of shape functions $\phi_i^j(x, z)$, $j \in \{1, ..., d_i\}$ spanning this space. The choice of the approximation space is local. In the numerical tests, I use standard polynomial approximation spaces P^k containing all polynomials of degree at most k with $k \in \{1, 2, 3\}$ (Cockburn and Shu, 1998).

Following Cheng and Wang (2014), I define $\boldsymbol{n}_{K_i}$ to be the outward unit normal to the K_i cell boundary and $\boldsymbol{t}_{K_i}$ the unit tangential vector. At cell interfaces, traces $v_h^{\pm}$, jumps $[v_h]$ and means $\overline{v_h}$, of any numerical quantity v_h defined inside two neighboring cells are given respectively by

$$\begin{aligned} v_h^{\pm}(\boldsymbol{x}) &= \lim_{\epsilon \downarrow 0} v_h(\boldsymbol{x} \pm \epsilon \boldsymbol{n}_{K_i}), \\ [v_h](\boldsymbol{x}) &= v_h^{+}(\boldsymbol{x}) - v_h^{-}(\boldsymbol{x}), \\ \overline{v_h}(\boldsymbol{x}) &= \frac{1}{2}\big(v_h^{+}(\boldsymbol{x}) + v_h^{-}(\boldsymbol{x})\big). \end{aligned} \tag{2.19}$$

At cell interface, a two-component vector is defined by the expression

$$\nabla_{\boldsymbol{x}} u_{h_{K_i}}^{\pm} = \left((\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i})^{\pm}, \overline{\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{t}_{K_i}}\right). \tag{2.20}$$

The first component $(\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i})^{\pm}$ is the projection onto the normal $\boldsymbol{n}_{K_i}$, of the gradient of the numerical solution computed inside the K_i cell $(-)$, or inside its neighbor $(+)$. The second component $\overline{\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{t}_{K_i}}$ is the mean of the projections onto the tangential vector $\boldsymbol{t}_{K_i}$ of the gradient of the numerical solution computed inside the K_i cell and inside its corresponding neighbor.

The weak formulation of equation (2.1) can be stated as follows:

Find $u_h(., \xi) \in \{v : v|_{K_i} \in \mathcal{P}_i, \forall i \in \{1, ..., n\}\}$ $\forall \xi \geqslant 0$ such that

$$\begin{aligned} &\int_{K_i} \Big(\partial_\xi u_h(\boldsymbol{x}, \xi) + \mathcal{H}\big(\boldsymbol{x}, \nabla_{\boldsymbol{x}} u_h(\boldsymbol{x}, \xi)\big)\Big) v_i(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &+ \int_{\partial K_i} \min\big(\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}, \xi), 0\big) [u_h](\boldsymbol{x}, \xi) v_i^{-}(\boldsymbol{x}) \mathrm{ds} \\ - C\Delta K_i \sum_{S_i^j \in \partial K_i} \frac{1}{\Delta S_i^j} &\int_{S_i^j} \big(\chi_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}, \xi) - |\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}, \xi)|\big) [\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i}](\boldsymbol{x}, \xi) v_i^{-}(\boldsymbol{x}) \mathrm{ds} = 0, \end{aligned} \tag{2.21}$$

for each $i \in \{1, ..., n\}$, and for any test function $v_i \in \mathcal{P}_i$.

Here, ΔK_i (respectively ΔS_i^j) is the size of the element K_i (respectively the length of the edge j of element K_i). The set ∂K_i denotes the internal edges of element K_i which are shared with other cells. The test functions v_i are shape functions as usual for Galerkin approaches (Zienkiewicz and Morgan, 2006).

In scheme (2.21), the following quantities are introduced:

$$\mathcal{H}_{K_i}^{\pm} = \mathcal{H}\big(\boldsymbol{x}^{\pm}, \nabla_{\boldsymbol{x}} u_{h_{K_i}}^{\pm}\big), \tag{2.22a}$$

$$\mathcal{H}_{\boldsymbol{n}_{K_i}} = \nabla_{\nabla u}\mathcal{H} \cdot \boldsymbol{n}_{K_i}, \tag{2.22b}$$

$$\mathcal{H}_{\boldsymbol{n}_{K_i}}^{\pm} = \mathcal{H}_{\boldsymbol{n}_{K_i}}\big(\boldsymbol{x}^{\pm}, \nabla_{\boldsymbol{x}} u_{h_{K_i}}^{\pm}\big), \tag{2.22c}$$

$$\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}) = \begin{cases} \frac{\mathcal{H}_{K_i}^{+} - \mathcal{H}_{K_i}^{-}}{[\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i}](\boldsymbol{x})}, & \text{if } [\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i}](\boldsymbol{x}) \neq 0, \\ \frac{1}{2}\left(\mathcal{H}_{\boldsymbol{n}_{K_i}}^{+} + \mathcal{H}_{\boldsymbol{n}_{K_i}}^{-}\right), & \text{otherwise,} \end{cases} \tag{2.22d}$$

$$\delta_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}) = \max\big(0, \tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}) - \mathcal{H}_{\boldsymbol{n}_{K_i}}^{-}, \mathcal{H}_{\boldsymbol{n}_{K_i}}^{+} - \tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x})\big), \tag{2.22e}$$

$$\chi_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}) = \max\big(\delta_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}), |\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x})|\big). \tag{2.22f}$$

The first term of scheme (2.21) ensures consistency. It embeds a weak formulation of the Hamilton–Jacobi partial differential equation inside each element. The scheme accounts for causality thanks to the quantity $\tilde{\mathcal{H}}_{\boldsymbol{n}_K}$, referred to as the Roe speed, estimated across interfaces between elements. This Roe speed is an extension of the group velocity vector $\partial\mathcal{H}/\partial\boldsymbol{p}$ to the numerical discontinuous case at interfaces of DG cells. Its sign determines the information flow direction at an interface. At a point located at such an interface between two cells i and j, we have

$$\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}} = -\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_j}}. \tag{2.23}$$

The second term of scheme (2.21) penalizes the jump of the solution at the interface, $[u_h]$, only in the cells where the Roe speed is negative

$$\tilde{\mathcal{H}}_{\boldsymbol{n}_K} < 0. \tag{2.24}$$

In other words, the *downwind* cell receives information from the *upwind* cell, and the *upwind* cell is not influenced by the *downwind* cell. When flows from both cells, determined by $\mathcal{H}_{\boldsymbol{n}_K}$, are oriented towards the other cell, the situation is equivalent to a shock in the case of hyperbolic conservation laws. In this case, which occurs next to wavefield triplications, the second terms captures the shock and keeps the smallest traveltime. In the opposite case where flows from both cells are inward, this is similar to what is called a rarefaction (see appendix in Qian and Symes, 2001). The entropy condition is violated in such cells.

Only in this case, the third term of scheme (2.21), referred to as the viscosity term, is non-zero. This yields a penalization on the jump of the normal component of spatial derivatives of the solution at the interface, $[\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i}]$, so as to correct the entropy violation. Thanks to this third term, the entropy fix is directly embedded in the scheme. This viscosity term is weighted by a ratio related to the geometry of the element, and balanced by an empirical constant C. Based on the recommendation of Cheng and Wang (2014), I have verified that the choice $C = 0.25$ yields satisfactory results in my numerical experiments.

hp-adaptivity Two interesting features of the DG method are the so-called h-adaptivity and p-adaptivity. The first one, h-adaptivity, shared with classical continuous finite-element methods, refers to the use of non-regular or unstructured meshes. Unlike finite-differences, finite-element methods do not require a regular Cartesian discretization of the domain. Triangles and/or rectangles might be used, and element size might vary. This can be attractive for several purposes:

- Mesh tailoring with respect to geometrical constraints, such as topography (see e.g. section 2.4.2, and chapter 3), or specific structures inside the model;

- Mesh refinement, in order to describe more precisely some specific parts of the solution (discontinuities, steep gradients, shocks, etc.). This refinement can be performed a priori, or iteratively: a solution is first computed in a regular mesh, a new mesh is built based on characteristics of the solution, and a new computation is performed. The adaptive mesh refinement (AMR) process for dynamic problems consists of refining the mesh during the computation, based on local error estimations made at each time step. Simulations in fields like electromagnetics, (magneto)hydrodynamics, weather and climate, astrophysics, make intensive use of AMR (Plewa et al., 2003).

However, one has to keep in mind that a non-regular mesh implies a certain level of book-keeping: the mesh characteristics must be kept in memory. In comparison, a regular Cartesian grid can be described only by its origin coordinates, the spacing value in each dimension, and the number of points/cells in each dimension. In the next chapter, I show that, for the fast-sweeping method, it is essential to keep regularity in the mesh, and I develop a simple deformation strategy, which is a good compromise between adaptivity, memory requirement for mesh storage in 3D, and specific constraints related to FSM.

As mentioned before, the choice of the approximation space where the numerical solution lies is attached to a given element. Therefore, different spaces might be assigned to different elements. This property is referred to as p-adaptivity in the literature, an interesting feature of the DG strategy for adjusting the numerical accuracy locally. If there is a special interest for particular areas of the solution, it might be appropriate to use higher-order approximation spaces in these areas, in order to describe more precisely the solution. However, this again implies to keep in memory the characteristics of the approximation space of each element.

Indeed, both h- and p-adaptivity can be combined in various ways. I have not investigated much on the use of p-adaptivity in my work, since I mainly focused on developing efficient and accurate solvers general enough to be suitable for various applications in geophysics. The main concern is related to topography, which I address with h-adaptivity.

Time integration Decomposing the numerical solution over each element K_i in terms of degrees of freedom u_i^j yields

$$u_{h|K_i}(x,z,\xi) = \sum_{j=1}^{d_i} u_i^j(\xi)\phi_i^j(x,z). \tag{2.25}$$

Replacing u_h by expression (2.25) in scheme (2.21) yields a system of ODE in $\partial_t u_i^j(\xi)$, namely the pseudo-time derivatives of the degrees of freedom. Time integration is then performed with a standard explicit Runge–Kutta (RK) scheme. Since we are looking for the steady-state, there is no need for a high level of accuracy on the intermediate states, so that a second-order RK scheme is enough. In my numerical experiments, I observe that using a first-order Euler explicit scheme yields numerical instability, while using higher-order RK schemes do not modify the steady state while increasing the computational cost. In practice, the steady state is detected by comparing the relative evolution of the solution between two successive time steps.

Implementation For the numerical implementation of the DG method, the edge and volume integrals are computed in a reference element with a suitable mapping. They are estimated using efficient quadrature rules with enough accuracy together with associated Gauss points, similar to Cockburn and Shu (1998). The integral computation is detailed in appendix 2.C. I employ modal basis functions inspired by Piperno (2005). Details regarding the basis functions are given in appendix 2.D.

In the numerical examples of sections 2.3 and 2.4, I use both triangular meshes (for topography) and rectangular meshes. The rectangular case writes like (2.21), but several simplifications are possible in the implementation: the edges are parallel to $x-$ or $z-$axis, so that projections onto normal and tangent vectors are straightforward.

The DG scheme is designed for any Hamilton–Jacobi equation, which makes it very attractive in the geophysical context since it can handle anisotropy as well as isotropy. The only changes to implement are related to the expression of the Hamiltonian, and its derivatives with respect to the derivatives of the solution, so as to compute the Roe speed. An additional step has to be integrated for tilted anisotropy, which is a projection of the numerical quantities from x- and z-axes onto the tilted local referential.

2.2.2 Adaptation to the geophysical problem

2.2.2.1 CFL condition and best Hamiltonian choice

Formulation of the CFL condition The stability of the RK time integration is constrained by a Courant–Friedrichs–Lewy (CFL) condition. This is a severe constraint in terms of computational cost since we are only interested in reaching the steady-state solution. Therefore, we should find ways to minimize this constraint. The CFL condition establishes a proportional relationship between the maximum size of time steps we can use and the characteristic length λ of the mesh. For a regular triangular mesh, this characteristic length λ is generally taken as the radius of the inscribed circle of a cell. For a rectangular Cartesian mesh, it is defined by half the length of the shortest edge of a cell. In my work, I have not investigated on the use of local time-stepping, which would introduce more complexity in the scheme. Therefore, time steps are global, which means that when using a non-regular mesh, we must consider the cell where the CFL constraint is the strongest, i.e. the cell having the smallest λ. This strongest constraint will determine the time step of the global RK integration. The CFL constraint also depends on the polynomial degree k we use for numerical approximation inside cells. We may write the CFL constraint into a general form

$$\Delta\xi \leqslant \frac{1}{2k+1}\lambda Q \tag{2.26}$$

The $\frac{1}{2k+1}$ factor comes from the analysis performed in the case of hyperbolic conservation laws (Cockburn and Shu, 1989).

The Q factor depends on the Hamiltonian we use, and connects the constraint on the time step to the numerical propagation velocity of the solution. In other words, we need to ensure that the numerical solution is able to propagate as fast as the physical one. The stability condition for RK-DG schemes can be established in the case of hyperbolic conservation laws (see e.g. Cockburn and Shu, 1989). These laws write in 1D

$$u_\xi + (f(u))_{,x} = 0. \tag{2.27}$$

The Q factor is given by

$$Q = \frac{\alpha}{\max|f'(u)|}, \tag{2.28}$$

where α is a constant. This can be extended to the case of Hamilton–Jacobi equations in one dimension with a convex Hamiltonian (Cheng and Shu, 2007; Cheng and Wang, 2014). We must consider the quantities $\max|\tilde{\mathcal{H}}_{K_i}|$ at cell boundaries and $\max|\partial\mathcal{H}(x,u_{,x})/\partial u_{,x}|$ inside cells, yielding

$$Q_{\text{1D}} = \frac{\alpha}{\max\left(\max|\partial\mathcal{H}(x,u_{,x})/\partial u_{,x}|\ ,\ \max|\tilde{\mathcal{H}}_{K_i}|\right)}. \tag{2.29}$$

For 2D and/or non-convex Hamiltonians, I have not found general proofs but I rely on the behaviors observed in my numerical tests. In a 2D case, the vector $\boldsymbol{\mathcal{H}}$ must be considered, the components of which are the derivatives of the Hamiltonian with respect to both components of the gradient of u, namely

$$\begin{aligned}\mathcal{H}_1(x,z,u_{,x},u_{,z}) &= \frac{\partial\mathcal{H}(x,z,u_{,x},u_{,z})}{\partial u_{,x}},\\ \mathcal{H}_2(x,z,u_{,x},u_{,z}) &= \frac{\partial\mathcal{H}(x,z,u_{,x},u_{,z})}{\partial u_{,z}}.\end{aligned} \tag{2.30}$$

Note that this vector is similar to the quantity $\partial\mathcal{H}/\partial\boldsymbol{p}$ used in section 1.3 and has a strong link with the group velocity vector as defined in equation (1.33). A criterion for defining Q in the 2-D case is given by Zhang et al. (2005c) in a finite-difference framework, using $\mathcal{H}_1$ and $\mathcal{H}_2$ as well as characteristic lengths of the grid along both x and z directions. However, since we want to use unstructured meshes, we need to keep a general criterion and consider the norm of the vector $\boldsymbol{\mathcal{H}}$, instead of considering x and z directions as well as $\mathcal{H}_1$ and $\mathcal{H}_2$ separately as in Zhang et al. (2005c). This yields

$$Q_{\text{2D}} = \frac{\alpha}{\max\left(\max||\boldsymbol{\mathcal{H}}||\ ,\ \max|\tilde{\mathcal{H}}_{K_i}|\right)}. \tag{2.31}$$

This norm is considered for instance by Qian and Symes (2002b) for the CFL derivation in a paraxial approach, from where we find $\alpha = \sqrt{2}/2$ in the general 2D case.

In our DG approach, we might also be careful about the quantity $\max|\tilde{\mathcal{H}}_{\mathbf{n}_{K_i}}|$ at cell boundaries, involved in expressions (2.29) and (2.31). As mentionned in section 2.2.1, $\tilde{\mathcal{H}}_{\mathbf{n}_{K_i}}$ is an extension of $\boldsymbol{\mathcal{H}}$ at the interfaces. In the isotropic case, it can be shown from the definitions (2.22) that values of $|\tilde{\mathcal{H}}_{\mathbf{n}_{K_i}}|$ at cell boundaries are bounded by those of $\mathcal{H}_1$ and $\mathcal{H}_2$ in neighbouring cells. I make the assumption that this also holds in the TTI case. The factor Q then simplifies to the expression

$$Q_{\text{2D}} = \frac{\alpha}{\max||\boldsymbol{\mathcal{H}}||}. \tag{2.32}$$

Isotropic case Isotropic Hamiltonian (2.3) gives the expression

$$\|\boldsymbol{\mathcal{H}}\| = 1 \quad \forall\mathbf{x}, \tag{2.33}$$

whereas Hamiltonian (2.5) yields

$$\|\boldsymbol{\mathcal{H}}\| = c(\mathbf{x}). \tag{2.34}$$

The factor Q in equation (2.26) is inversely proportional to the maximum of these quantities. We see that, in both cases, Q does not depend on the solution u nor its derivatives $u_{,i}$. If we consider a heterogeneous medium, the global constraint on time steps in (2.34) is imposed by the highest speed value since

$$Q = \frac{\alpha}{\max\limits_{\Omega} c(\mathbf{x})}. \tag{2.35}$$

This maximum speed value occurs only in a subdomain of Ω, so that the computation is not optimal in terms of number of iterations. For this reason, although (2.5) has the meaning of mimicking the wavefront evolution in the physical medium from the source, I prefer to use (2.3) for computational efficiency. The CFL constraint is uniform in space in this formulation, due to the uniform propagation velocity of the solution. I illustrate the computational advantage of this choice in section 2.3.1.

2D TI case Performing the same analysis is less straightforward in the anisotropic case. For the sake of clarity, I first consider the VTI case. The differentiation of the VTI Hamiltonian (1.44) yields

$$\begin{aligned} \mathcal{H}_1 &= 2du_{,x} + 2cu_{,x}u_{,z}^2, \\ \mathcal{H}_2 &= 2eu_{,z} + 2cu_{,x}^2u_{,z}, \end{aligned} \tag{2.36}$$

$$\|\boldsymbol{\mathcal{H}}\| = 2\sqrt{d^2u_{,x}^2 + e^2u_{,z}^2 + \big(2c(d+e) + c^2(u_{,x}^2 + u_{,z}^2)\big)u_{,x}^2u_{,z}^2}. \tag{2.37}$$

The VTI Hamiltonian is not Lipschitz continuous. The value of $\|\boldsymbol{\mathcal{H}}\|$ depends on the derivatives of the solution $u_{,x}$ and $u_{,z}$ in an unbounded way. This is not desirable because we cannot assign a value to $\Delta\xi$ in (2.26) since the Q factor is virtually equal to zero, in view of equation (2.32). For this reason I switch to another VTI Hamiltonian, which yields the same steady state as (1.44). The new VTI Hamiltonian is similar to the one given by Zhang et al. (2006, eq. 3.12) and writes

$$\mathcal{H}_{VTI} = \frac{1}{\sqrt{d}}\left(\sqrt{\sqrt{\frac{1}{4}\left(du_{,x}^2 + eu_{,z}^2\right)^2 + cu_{,x}^2u_{,z}^2} + \frac{1}{2}\left(du_{,x}^2 + eu_{,z}^2\right)} - 1\right). \tag{2.38}$$

I demonstrate the equivalence of Hamiltonians (1.44) and (2.38) at steady state in appendix 2.A. As in the case of isotropic Hamiltonian (2.3), this new VTI Hamiltonian (2.38) is Lipschitz continuous of Lipschitz constant 1, giving a predictable stable $\Delta\xi$ for integration. I demonstrate this property in appendix 2.B. The TTI case is retrieved by introducing the tilt angle θ as in (1.46). Note that the factorization strategy for Hamiltonian (2.38) is the same as for Hamiltonian (1.44). It is now possible to estimate a time step $\Delta\xi$ for performing stable computation in arbitrary heterogeneous TTI media.

I have shown how to optimize the CFL constraint in both isotropic and anisotropic cases. I emphasize that the use of the additive factorization technique does not change any of the above conclusions.

2.2.2.2 Boundary conditions

Scheme (2.21) is designed for general Hamilton–Jacobi equations with no specific boundary conditions. Typically, the illustrations given in Cheng and Wang (2014) are performed with periodic boundary conditions. However, in a geophysical setting, we need to simulate wave propagation and associated traveltime in a finite computational domain (a part of the subsurface) embedded in a larger physical domain (the Earth's interior). The traveltime computation inside the domain must not be polluted by information coming from the outside. For this, I design a suitable boundary condition as an additional flux term in the DG scheme, which applies only at external boundaries. The principle is the following: information flux at interfaces is given by the Roe speed $\tilde{\mathcal{H}}_{\boldsymbol{n}_K}$. Since no information must enter the domain from the outside, we need to verify the positivity of this quantity at external boundaries. If this quantity is negative, a penalization is introduced in the same way as in the third term of scheme (2.21), namely the entropy penalization term. This yields the following new scheme:

Find $u_h(.,\xi) \in \{v : v|_{K_i} \in \mathcal{P}_i, \forall i \in \{1,...,n\}\}\ \forall \xi \geqslant 0$ such that

$$\begin{aligned}
&\int_{K_i} \Big(\partial_\xi u_h(\boldsymbol{x},\xi) + \mathcal{H}\big(\boldsymbol{x}, \nabla_{\boldsymbol{x}} u_h(\boldsymbol{x},\xi)\big)\Big) v_i(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\
&+ \int_{\partial K_i} \min\big(\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi), 0\big) [u_h](\boldsymbol{x},\xi) v_i^-(\boldsymbol{x}) \mathrm{ds} \\
- C\Delta K_i \sum_{S_i^j \in \partial K_i} \frac{1}{\Delta S_i^j} &\int_{S_i^j} \big(\chi_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi) - |\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi)|\big) [\nabla_{\boldsymbol{x}} u_h \cdot \boldsymbol{n}_{K_i}](\boldsymbol{x},\xi) v_i^-(\boldsymbol{x}) \mathrm{ds} \\
- 2C\Delta K_i \sum_{\bar{S}_i^j \in \bar{\partial} K_i} \frac{1}{\Delta \bar{S}_i^j} &\int_{\bar{S}_i^j} \min\big(\mathcal{H}^-_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi), 0\big)\ (\nabla_{\boldsymbol{x}} u_h^-(\boldsymbol{x},\xi) \cdot \boldsymbol{n}_{K_i})\ v_i^-(\boldsymbol{x})\ \mathrm{ds} = 0,
\end{aligned} \tag{2.39}$$

for each $i \in \{1,...,n\}$, and for any test function $v_i \in \mathcal{P}_i$.

The set $\bar{\partial} K_i$ denotes the external edges of element K_i which are part of the domain boundary $\partial\Omega$. Since there is no neighbor for an external edge, the Roe speed is evaluated by the quantity $\mathcal{H}^-_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi)$. If it is negative, a penalization is introduced on $\nabla_{\boldsymbol{x}} u_h^-(\boldsymbol{x},\xi) \cdot \boldsymbol{n}_{K_i}$, instead of the jump of this quantity as in the third term of the scheme, since there is no external neighbor. This is the reason why a factor 2 is introduced before this additional term.

The additional term in scheme (2.39) is referred to as an outgoing flux term, or a radiative boundary condition. Let me add two remarks:

1. This term is suitable for boundaries of the domain at depth, but also for free-surface (topography) boundaries;
2. Propagation along boundaries is permitted. This can occur along the surface as well as at depth. However, the area affected by such propagation at depth should be considered with care: it sometimes mean that the model should be extended, or that the traveltime solution in this area is not physically meaningful.

2.2.2.3 Point-source and initial conditions

A condition must be implemented at the source point. This is indeed a boundary condition, since it is located in space and it holds for every pseudo-time step until the steady state is reached. At the source point, the traveltime value is zero. The source location falls into one element of the mesh, referred to as the source element. All the degrees of freedom of this source element are set to pre-computed values, which are kept unchanged during the computation process. These values correspond to a projection of a specified solution around the source into the approximation space of the source element. When we use the factorization technique, then the solution τ, computed as a perturbation to the reference solution u_0, is set to zero inside the source element, since u_0 is supposed to be a suitable approximation in the vicinity of the source.

In addition to the zero-traveltime condition, a correct implementation of the source condition should ensure an outgoing flux from the source element towards its neighbors. This is naturally enforced by the reference solution u_0. However, if the source point is located near an interface between elements,

or near a corner of element, then the spatial derivatives of the reference solution u_0 at interfaces with neighbors, involved in the flux terms of scheme (2.39), might not be well oriented. In this case, if the source cannot be relocated, then I employ an appropriate strategy which extends the source area. I consider two or four neighboring elements as source elements. Inside these elements I set the source condition $\tau(\boldsymbol{x}, \xi) = 0$, and they are not updated during the computation.

An initial solution must be set everywhere, and the simplest case is to consider u_0 as an initial value for traveltime in the whole medium. Therefore, I use the following initial condition:

$$\tau(\boldsymbol{x}, 0) = 0. \tag{2.40}$$

I have verified that the choice of a different initial condition does not modify the steady state, nor the number of pseudo-time steps required to reach it, nor the computational efficiency. This is linked to the fact that the numerical solution expands from the source in the medium with a propagation velocity which is independent from the current value of the solution.

2.2.3 Conclusion

Considering the original DG scheme (2.21) from Cheng and Wang (2014), I have developed several additional features, such as boundary conditions and point-source consideration (factorization, implementation as a boundary condition), which make the new scheme (2.39) suitable for geophysical problems.

In (2.39), the evolution of the degrees of freedom in time $\partial_t u_i^j(t)$ of element K_i can be computed without knowledge of any $\partial_t u_k^j(t), k \neq i$. Therefore, the RK-DG scheme is easy to parallelize in space as a block-Jacobi method: one can compute $\partial_t u_i^j(t)$ independently for each element K_i before updating the solution everywhere. In terms of implementation, this is done with OpenMP procedures in the prototype code I have developed.

Next, I present an extensive numerical validation (section 2.3) as well as some examples in various geophysical contexts (section 2.4).

2.3 Convergence analysis

In this section, I illustrate the convergence properties of the RK-DG scheme (2.39) in two smooth isotropic media where the exact solution is known. I focus on the ℓ^2 error computed on the solution and its derivatives, in order to perform systematic convergence analysis yielding the numerical orders of convergence with respect to the mesh refinement.

Inside a computation domain discretized with K elements sharing the same approximation space, the ℓ^2 error is computed as

$$\ell^2 \text{ error} = \sqrt{\frac{\sum_{i=1}^{K}\sum_{j=1}^{G}\left(u_h(\mathbf{x}_i^j) - u_a(\mathbf{x}_i^j)\right)^2}{\sum_{i=1}^{K}\sum_{j=1}^{G} u_a^2(\mathbf{x}_i^j)}}, \tag{2.41}$$

where the values of the numerical solution u_h are compared to those of the closed-form solution u_a at each of the G Gauss points of each of the K elements of the discretized domain. The ℓ^2 errors of the

x- and z-derivatives of the solution are also computed in the same way, replacing u_h and u_a by their x- and z-derivatives respectively in (2.41).

2.3.1 Smooth isotropic velocity

In this first case study, I analyze the convergence orders of the RK-DG scheme (2.39) in a simple isotropic medium for various settings:

- Source treatment: no treatment, enforcement of the exact solution on a fixed area around the source, point-source factorization;
- Polynomial orders: P^1, P^2, P^3;
- Meshes: triangular and Cartesian rectangular;
- Hamiltonians: both (2.3) and (2.5) are compared.

I also perform a comparison with the finite-difference isotropic Eikonal solver from Noble et al. (2014).

The computation is performed inside a 4×4 km square. Inside this domain, a constant vertical gradient is defined for the speed $c(z)$ (km.s^{-1}), such that

$$c(z) = 1 + 0.5z. \tag{2.42}$$

The point source is located in the center ($x_s = 2$ km, $z_s = 2$ km). The solution of the Eikonal equation is known in closed form in such a case (Fomel et al., 2009), for a source velocity $c(\boldsymbol{x}_s) = c_0$ and a gradient of velocity $\boldsymbol{G}_0$:

$$T(\boldsymbol{x}) = \frac{1}{|\boldsymbol{G}_0|}\text{arccosh}\big(1 + \frac{|\boldsymbol{G}_0|^2|\boldsymbol{x} - \boldsymbol{x}_s|^2}{2c(\boldsymbol{x})c_0}\big). \tag{2.43}$$

It is then straightforward to obtain spatial derivatives of $T(\boldsymbol{x})$. Isocontours of the exact traveltime solution are shown in figure 2.1 together with the velocity model.

The domain is first discretized in a rectangular Cartesian frame with $N_x = N_z = N$ elements along x- and z- axes so that the total number of elements is N^2.

No point-source: Optimal convergence orders As mentioned in section 2.1.2.1, it is possible to avoid the point-source singularity when studying convergence orders by imposing a grid-independent constant-velocity area. This yields optimal convergence orders. In order to illustrate the expected convergence behavior of the RK-DG solver, I perform a first set of simulations with an extended source area. This area contains all the elements distant of less than 0.4 km from the source point. These source elements are excluded from the computational domain, and the exact solution is used for computing their contribution in the integrals at the edges shared with the elements within the computational domain. I use Hamiltonian (2.3) without factorization. Figure 2.2 illustrates the optimal $k + 1$ convergence orders reached for P^1, P^2 and P^3 approximation spaces regarding the ℓ^2 error of the numerical solution. The optimal k convergence orders for the x-derivative of the solution can also be observed. Here and after, I do not exhibit the results related to the z-derivative of the solution, since it always yields the same convergence orders and thus the same conclusions as for the x-derivative. Note that in all the convergence figures, the errors are plotted with respect to the number of degrees of freedom in a log-log scale. In 2D, the number of degrees of freedom is proportional to N^2. Therefore, convergence orders are given by twice the slopes of the convergence curves, because the mesh spacing is proportional to $1/N$.

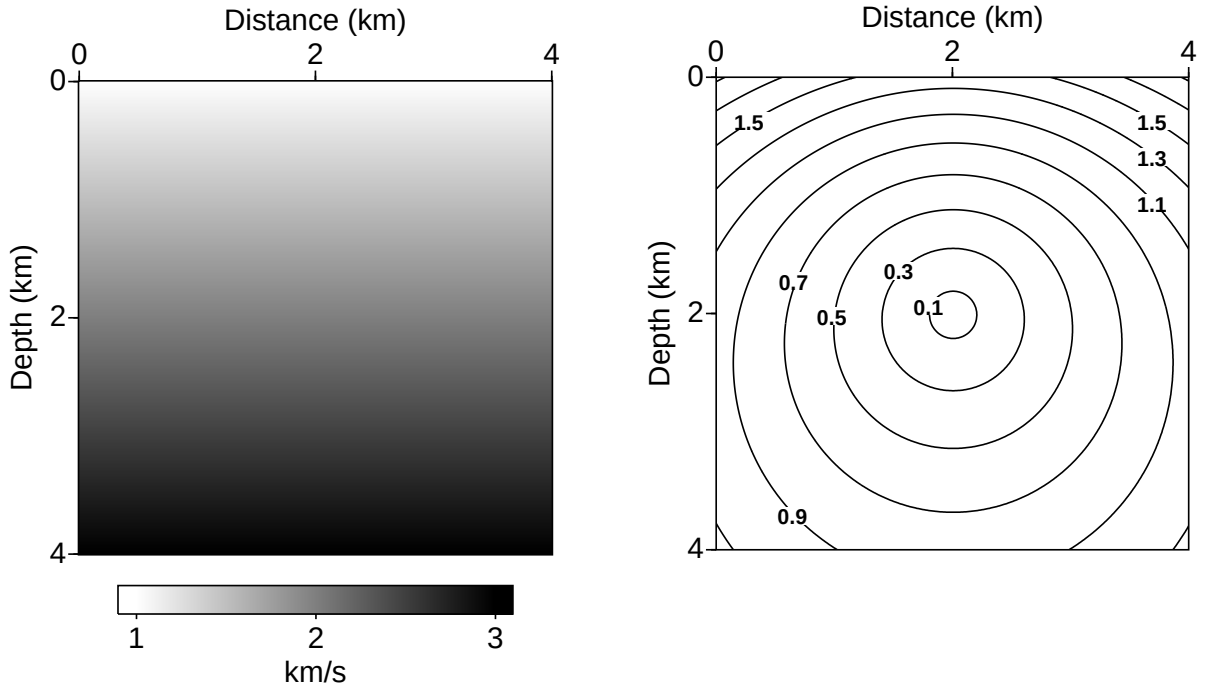


Figure 2.1: Medium with constant vertical gradient of velocity. Left: velocity model. Right: isocontours of the exact solution at different times (seconds).

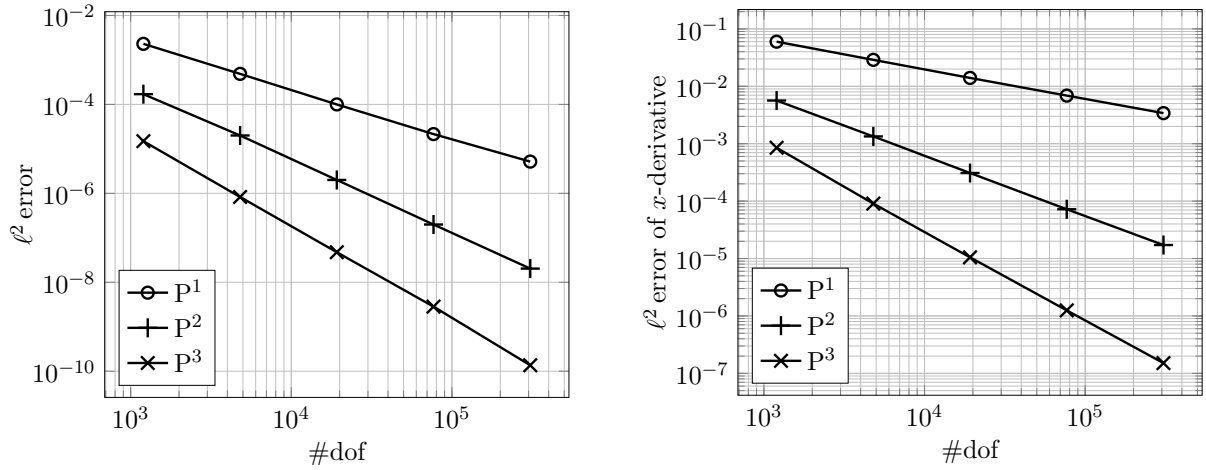


Figure 2.2: Medium with constant vertical gradient of velocity. ℓ^2 error of the solution (left) and ℓ^2 error of the x-derivative of the solution (right) with respect to the number of degrees of freedom (#dof) in the Cartesian rectangular case for P^1, P^2 and P^3 polynomial approximations, with a source area of radius 0.4 km: optimal $k+1$ convergence of the solution and k convergence of its x-derivative in a setting with no point source.

Point-source singularity: Non-optimal convergence orders In a realistic application, the exact solution around the source might not be obtained in closed form, so that the previous treatment is not applicable. The above convergence behavior is expected to collapse when the point-source singularity is introduced. If no special treatment is performed at the source point, when using Hamiltonian (2.3), a

	P^1			P^2			P^3		
N	#dof	ℓ^2 error	Order	#dof	ℓ^2 error	Order	#dof	ℓ^2 error	Order
20	1200	$4.84\mathrm{E}-03$		2400	$1.77\mathrm{E}-03$		4000	$7.18\mathrm{E}-04$	
40	4800	$2.24\mathrm{E}-03$	1.11	9600	$8.60\mathrm{E}-04$	1.04	16000	$2.18\mathrm{E}-04$	1.72
80	19200	$1.09\mathrm{E}-03$	1.03	38400	$4.29\mathrm{E}-04$	1.00	64000	$8.17\mathrm{E}-05$	1.41
160	76800	$5.45\mathrm{E}-04$	1.01	153600	$2.15\mathrm{E}-04$	1.00	256000	$3.69\mathrm{E}-05$	1.15
320	307200	$2.72\mathrm{E}-04$	1.00	614400	$1.08\mathrm{E}-04$	1.00	1024000	$1.80\mathrm{E}-05$	1.04

Table 2.1: Medium with constant vertical gradient of velocity. Number of degrees of freedom (#dof), ℓ^2 error of the solution and convergence orders for several values of N in the Cartesian rectangular case for P^1, P^2 and P^3 polynomial approximations, without factorization: non-optimal first-order-only convergence due to the source singularity.

	P^1			P^2			P^3		
N	#dof	ℓ^2 error	Order	#dof	ℓ^2 error	Order	#dof	ℓ^2 error	Order
20	1200	$5.24\mathrm{E}-04$		2400	$1.30\mathrm{E}-04$		4000	$2.21\mathrm{E}-05$	
40	4800	$1.27\mathrm{E}-04$	2.05	9600	$3.30\mathrm{E}-05$	1.98	16000	$5.59\mathrm{E}-06$	1.98
80	19200	$3.11\mathrm{E}-05$	2.03	38400	$8.29\mathrm{E}-06$	1.99	64000	$1.41\mathrm{E}-06$	1.99
160	76800	$7.71\mathrm{E}-06$	2.01	153600	$2.08\mathrm{E}-06$	2.00	256000	$3.52\mathrm{E}-07$	2.00
320	307200	$1.92\mathrm{E}-06$	2.01	614400	$5.20\mathrm{E}-07$	2.00	1024000	$8.81\mathrm{E}-08$	2.00

Table 2.2: Medium with constant vertical gradient of velocity. Number of degrees of freedom (#dof), ℓ^2 error of the solution and convergence orders for several values of N in the Cartesian rectangular case for P^1, P^2 and P^3 polynomial approximations, with factorization: second-order convergence is achieved, which is optimal in the P^1 case.

non-optimal first-order-only convergence of the solver is expected, whatever the degree of polynomials used. This is exhibited in table 2.1 and in figure 2.3. However, even if the convergence order is the same, there is still a gain in accuracy when increasing the polynomial order: the error magnitude decreases when the polynomial order increases for a given N. This is also the case for a given number of degrees of freedom. The spatial derivatives of the solution exhibit a degenerated first-order convergence which is controlled by the first-order-only convergence of the solution itself.

Point-source factorization: Increasing the convergence orders When the factorization technique is applied, which means I use Hamiltonian from equation (2.10) instead of (2.3) and the reference solution from equation (2.14), a gain of one order of convergence is observed, as shown in table 2.2. For P^1, P^2 and P^3 polynomial approximations a second-order accuracy is achieved, which is optimal for P^1. Here again, even if the convergence orders are the same, the error magnitude decreases when the polynomial order increases. An optimal first-order convergence for the x-derivative is observed in the P^1 case. Degenerated second-order convergences are observed for P^2 and higher-order polynomial spaces, dominated by the second-order convergence of the solution itself.

Conclusions of the two last paragraphs are synthesized in figure 2.3. The factorization technique leads to an optimal P^1 second-order solver as well as non-optimal second-order P^2 and P^3 solvers. The P^2 solver nearly reaches second-order optimality in terms of derivatives. In all the cases, the factorization yields a significant decrease of the error magnitude.

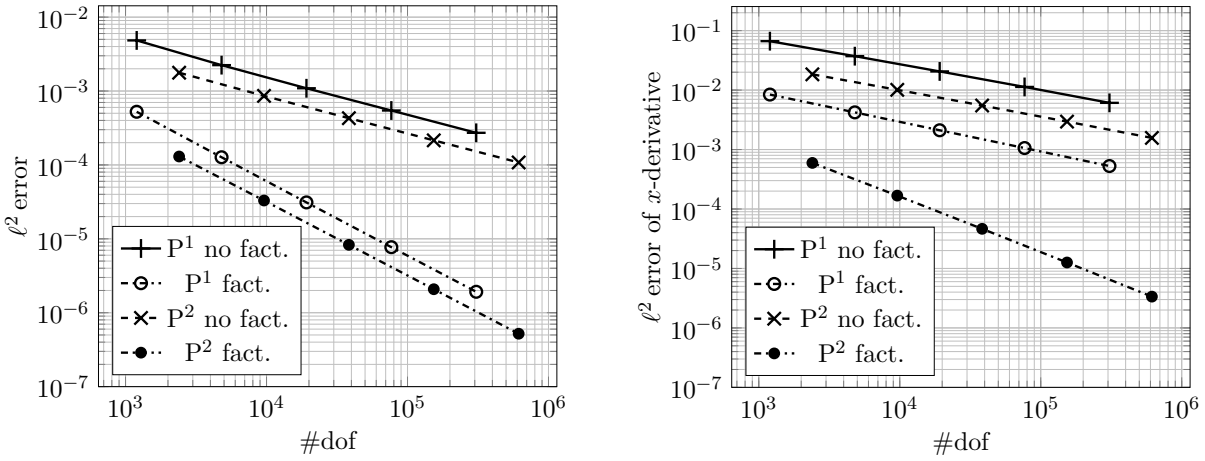


Figure 2.3: Medium with constant vertical gradient of velocity. ℓ^2 error of the solution (left) and ℓ^2 error of the x-derivative of the solution (right) with respect to the number of degrees of freedom (#dof) in the Cartesian rectangular case for P^1 and P^2 polynomial approximations, with factorization (fact.) and without factorization (no fact.). Left: non-optimal first-order convergence of the solver without factorization; second-order convergence with factorization (optimal for P^1). Right: degenerated first-order convergence of the x-derivative without factorization; optimal fully first-order convergence (P^1) and nearly second-order convergence (P^2) with factorization.

Triangular discretization Similar results are obtained when a structured triangulation of the domain is used. The Union-Jack (UJ) pattern is obtained from the Cartesian grid by cutting each rectangular element into two triangles in an alternating diagonal direction, as shown in figure 2.4. With $N_x = N_z = N$, the number of elements of this triangulation is now $2N^2$. Results are the same as in the Cartesian rectangular case in terms of convergence orders. However, the magnitude of error with respect to the number of degrees of freedom is higher, which is illustrated in the P^2 case in figure 2.5. This means that the UJ triangular case is less optimal than the rectangular case.

Hamiltonian choice As discussed in section 2.2.2.1, computations have been performed with Hamiltonian (2.3) as well as with the factorized formulation (2.10), instead of Hamiltonian (2.5) and its factorized formulation. I illustrate the advantage of this on a simple case study. Since the speed is not homogeneous in the present case study, a non-optimal CFL constraint is expected if Hamiltonian (2.5) or its factorized formulation is used. In the case of a P^2 Cartesian rectangular discretization with $N = 80$, which implies 38400 degrees of freedom, we observe that the computation takes 380 time integration steps to reach the steady state when using (2.3), whereas this number reaches 650 when using (2.5), as shown in table 2.3. The computational cost being proportional to this number of time steps, this is a clear illustration of the importance of choosing the best Hamiltonian for the steady-state problem.

Comparison with a finite-difference code Finally, I compare the results obtained in the Cartesian rectangular P^1 discretization with the factorization technique with those obtained with the 2D version of the fast-sweeping-based first-order finite-difference solver from Noble et al. (2014). The FSM-FD solver is based on local operators which also make use of the factorization principles. Figure 2.6 highlights the much lower error level of the DG solver compared to the one of the FSM-FD solver with

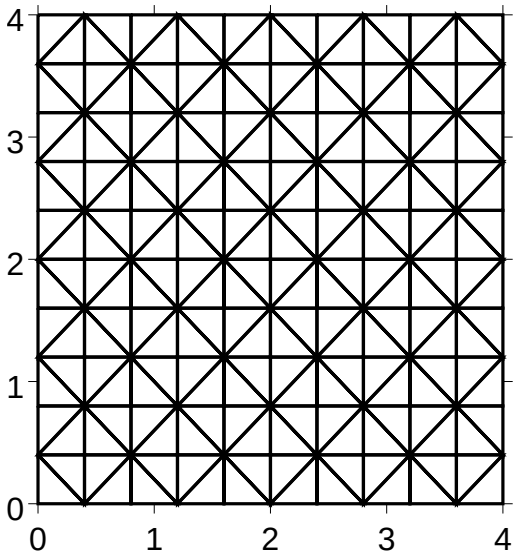


Figure 2.4: The Union-Jack triangulation shown for $N_x = N_z = 10$.

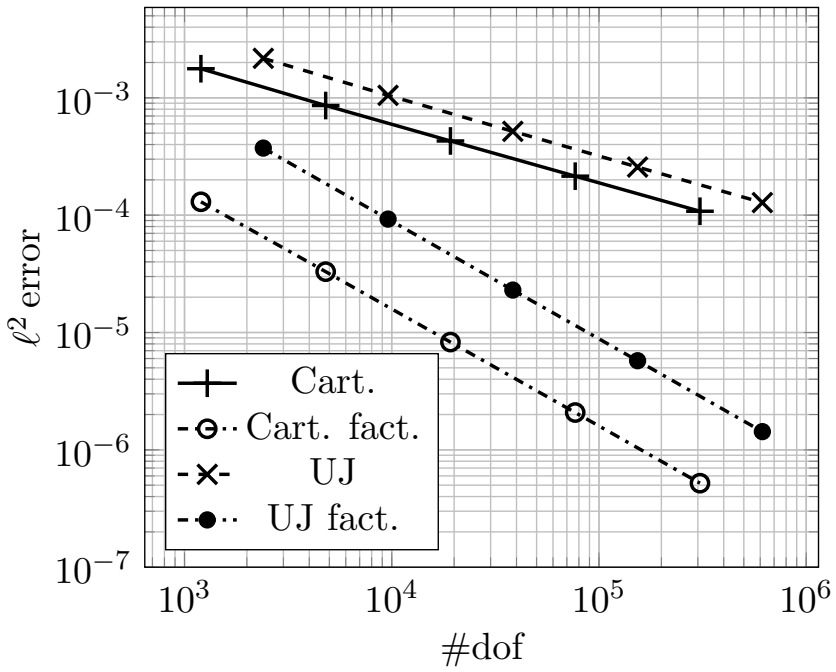


Figure 2.5: Medium with constant vertical gradient of velocity. ℓ^2 error of the numerical solution with P^2 polynomial approximations, in both Cartesian rectangular and UJ triangular cases, and both without factorization (Cart. and UJ) and with factorization (Cart. fact. and UJ fact.). The first-order convergence of the standard case is improved to a second-order convergence when the factorization is applied. Both Cartesian rectangular and UJ triangular discretizations achieve the same convergence orders, although the magnitude of error is higher in the UJ case.

N	steps Hamil. (2.3)	steps Hamil. (2.5)
80	380	650

Table 2.3: Medium with constant vertical gradient of velocity. Number of integration steps required to reach the steady state with a P^2 polynomial approximation, for $N = 80$, using Hamiltonians (2.3) and (2.5) with the factorization. In a heterogeneous medium, Hamiltonian (2.3) is more efficient.

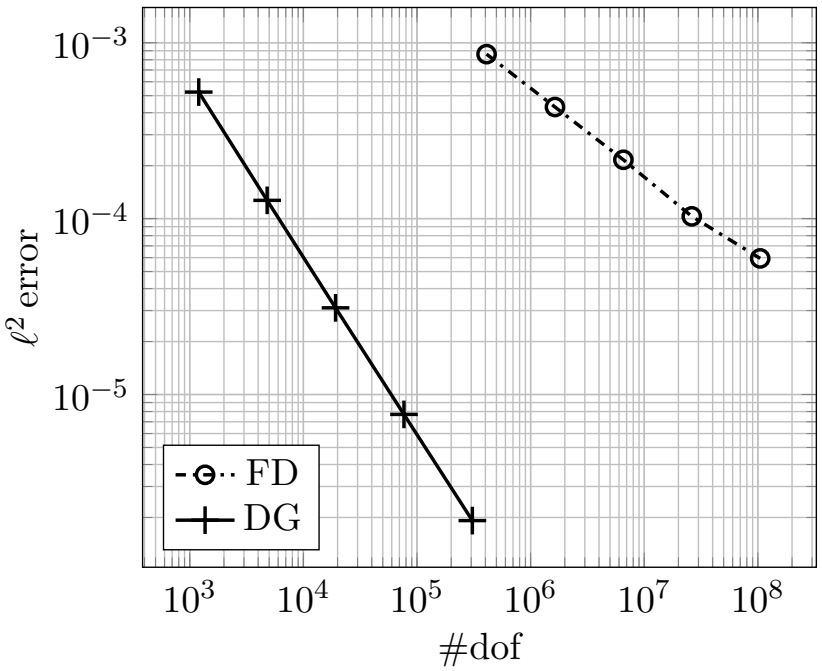


Figure 2.6: Medium with constant vertical gradient of velocity. ℓ^2 error of the numerical solution with the DG solver, P^1 approximation, Cartesian rectangular discretization and factorization (DG), compared to the FD solver from Noble et al. (2014).

respect to the number of degrees of freedom. The different slopes of the two lines exhibit the first-order convergence of the FSM-FD solver compared to the second-order convergence of the DG solver. The huge difference between the two lines has to be balanced by the fact that the FSM-FD solver is a faster solver, thanks to the use of a fast-sweeping strategy.

2.3.2 Higher-order factorization

The purpose of this section is to exhibit a way to gain one more order of convergence thanks to a higher-order source factorization. The principle holds in choosing a suitable reference solution $u_0(\mathbf{x})$ accounting for higher-order terms of the power series expansion of the velocity model at the source point. Luo et al. (2014b) proposed this kind of approach for the squared slowness and the squared Eikonal, as mentioned in section 2.1.2.2, equation (2.16).

In this simple example, the same domain of computation and source location as in the previous case are considered, but with a different speed. Slowness $s(\mathbf{x})$ is defined by

$$s(\mathbf{x}) = \frac{1}{c(\mathbf{x})}. \tag{2.44}$$

The medium is defined by a constant vertical gradient of the squared slowness (slowness unit s.km^{-1}):

$$s^2(z) = 0.25 - 0.1(z - z_s), \tag{2.45}$$

where the depth is expressed in km. The slowness is indeed strictly positive inside the domain. Again, the closed-form solution of the problem is known (see e.g. Fomel et al., 2009) as well as its spatial derivatives. We can write the speed as

$$c(z) = \frac{1}{\sqrt{0.25 - 0.1(z - z_s)}}, \tag{2.46}$$

which can be expanded around the source point as

$$c(z) = 2 + 0.4(z - z_s) + O\left((z - z_s)^2\right). \tag{2.47}$$

The standard factorization technique would account for the first term of the above expansion, using the solution in a constant velocity model of value $c_0 = 2\,\mathrm{km\,s^{-1}}$ as the reference solution. Here, I propose to account for both terms of the expansion using the exact solution in a constant gradient of velocity model as the reference solution. Therefore, in this example, $u_0(\mathbf{x})$ is the exact solution for the point-source problem in a velocity model

$$c_0(z) = 2 + 0.4(z - z_s), \tag{2.48}$$

with units of the equation (2.45). Computations are performed with P^1, P^2 and P^3 in Cartesian rectangular meshes and convergence behaviors are compatible with the order of selected polynomials (figure 2.7). Here again, I do not present results related to the z-derivative of the solution for the sake of concision, but it yields the same convergence orders and thus the same conclusions as the x-derivative.

This case study illustrates that this second-order factorization provides a third-order convergence. In a general case, for a given velocity model which can be expanded in power series at the source point, we are able to design a second-order reference solution based on the closed-form solution in a constant gradient of speed oriented in the direction of the gradient of the true velocity model. The resulting third-order scheme with a P^2 approximation is particularly advantageous for applications requiring quantities related to first- and second-order derivatives of the traveltime, such as amplitude and take-off angle (Luo et al., 2012, 2014a), or CRS and diffraction analysis which needs curvature information (e.g. Hubral, 1983; Dell and Gajewski, 2011; Schwarz et al., 2014; Bauer et al., 2017).

Remark Here, the reference solution u_0 is correctly defined inside the whole domain of computation since the corresponding speed c_0 in (2.48) is strictly positive. However, this is not true in the general case. For instance, in the same example, if the source is deeper than 5 km, then the formula (2.48) yields negative velocities at the surface. Other developments should be considered for an extensive work on higher-order factorization. See, for instance, the hybrid scheme designed in Luo et al. (2014b), in which the factored Eikonal equation is considered in a region surrounding the source, while the original equation is considered out of this region. I have not investigated more on that topic, since I no longer consider higher-order factorization in what follows. Standard factorization is sufficient for the applications that I test.

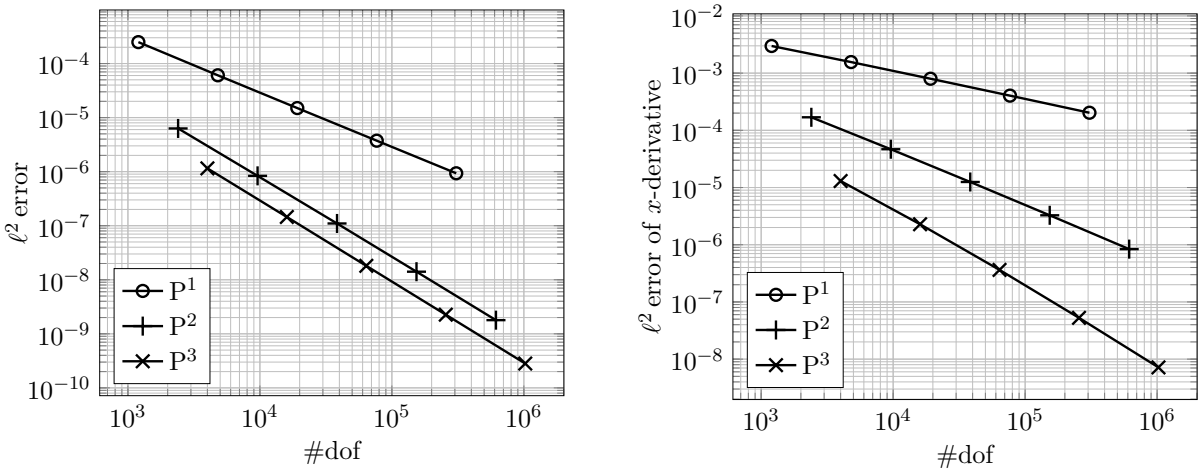


Figure 2.7: Medium with constant vertical gradient of slowness squared. ℓ^2 error of the solution (left) and ℓ^2 error of the x-derivative of the solution (right) with respect to the number of degrees of freedom (#dof) in the Cartesian case for P^1, P^2 and P^3 polynomial approximations, with a second-order factorization. Left: optimal second-order convergence (P^1); optimal third-order convergence (P^2); non-optimal third-order convergence (P^3). Right: optimal first-, second- and third-order convergences of the x-derivative for P^1, P^2 and P^3, respectively.

2.4 Numerical illustrations

I have validated the behavior of the RK-DG scheme in terms of convergence orders in simple isotropic media where the exact solution is known, and I have shown that the factorization technique is able to handle the point-source singularity in a satisfactory way. In this section, I exhibit some computation results obtained in more complex settings: anisotropy, non-flat topography, and highly heterogeneous realistic model. I give three numerical illustrations of the RK-DG solver in complex media:

1. **Homogeneous TTI velocity model**: Several approaches for point-source factorization are tested in presence of anelliptical anisotropy, and the results are compared with results from a finite-difference Eikonal solver detailed in Waheed et al. (2015b) and Tavakoli F. et al. (2015), as well as with an analytical solution;

2. **Non-flat topography**: A synthetic TTI model of a volcano is tested for various source locations;

3. **Realistic TTI model**: A complex heterogeneous model is tested for various source locations, and a comparison with a finite-difference code is performed.

2.4.1 Homogeneous TTI velocity model

In this illustration, I validate the ability of the RK-DG solver to compute traveltime in a TTI medium using the tilted formulation of Hamiltonian (2.38) as well as the corresponding factorized formulation. The medium is defined by homogeneous vertical speed, Thomsen's parameters and tilt angle, respectively $V_P = 2\,\mathrm{km\,s^{-1}}$, $\epsilon = 0.4$, $\delta = 0.2$ and $\theta = 40$ deg. The computation domain is a rectangle of 32 km length and 8 km depth, and the point source is located at $x = 2.025$ km, $z = 2.025$ km. For this

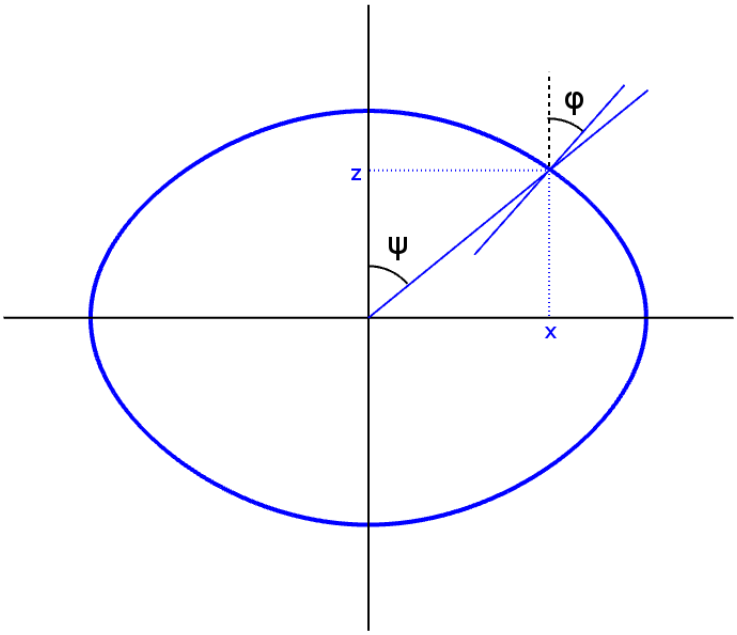


Figure 2.8: At any point (x, z) of a homogeneous TTI medium, the group (ray) angle ψ defined by x, z and the source point differs from the phase angle φ which is defined by the normal to the wavefront, except on the axes where they are equal. The parametric relationship between point coordinates and traveltime gives an explicit way to compute a group angle from a given phase angle. The inverse problem is solved iteratively.

anelliptical homogeneous medium, the exact solution is not known directly, but a parametric formulation gives an explicit way to compute the group angle (energy flux direction) and the position of the wavefront at a given time t for a given phase angle (normal to the wavefront, see figure 2.8). This formulation can be found in, e.g., Payton (1983, eqs. 2.8.8 and 2.8.9) and Carcione et al. (1988, eq. 5.9), and is given in appendix 2.E. Hence it is possible to build the exact wavefront at a given time *t* with a dense sampling of phase velocities (black line in figure 2.10), and visually compare the isocontours of the traveltime maps computed by the solver with these wavefronts.

The medium is discretized in a Cartesian frame with $N_x = 640$ and $N_z = 160$ and a P^1 approximation, yielding 307200 degrees of freedom. The first computation is performed without any factorization (red line in figure 2.10). The second computation uses the factorization technique with the elliptical reference solution from (2.17) and (2.18) (green line in figure 2.10). Since $\epsilon \neq \delta$, the anisotropy is anelliptical, so that the elliptical reference solution does not account for the whole source singularity. However, we observe an improvement of the solution compared to the first computation.

In a third computation, I pre-compute values for the exact anelliptical solution and its spatial derivatives at the points where they are required, and I use them as the reference solution. Since the reference solution and its derivatives are involved inside the integrals of scheme (2.39), their values are required at each Gauss point necessary for integral estimation by quadrature rules. I pre-compute these values by retrieving the phase angle from the group angle at a given point iteratively in a similar way as in Qian and Symes (2001, alg. 1), using the parametric formulation mentioned above (see figure 2.8), and a standard Newton method. The correct phase angle is retrieved at the computer precision in less than 8 iterations. I then approximate x- and z- derivatives using central differences. Since the wavefront has a smooth shape, these estimations are highly accurate. Once the required values are pre-computed and

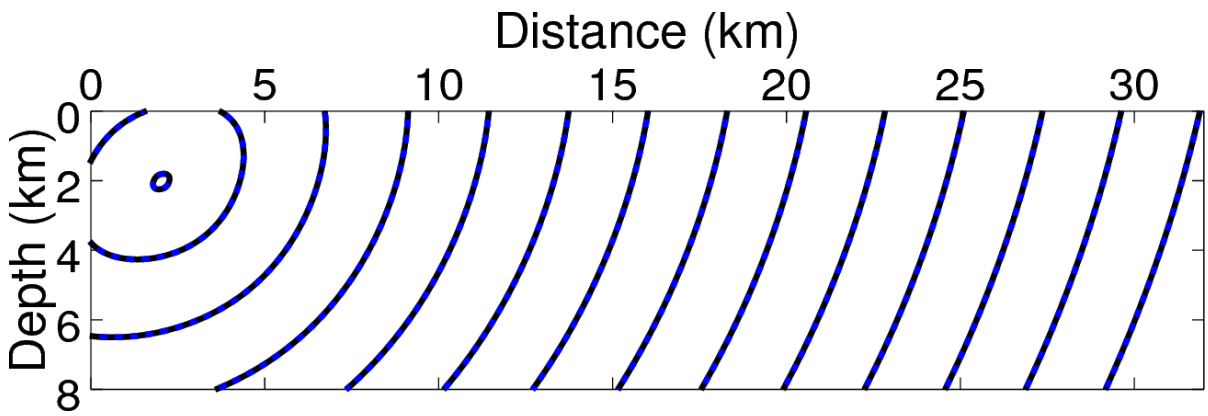


Figure 2.9: Isocontours of the solutions for the homogeneous TTI medium. Exact and numerical solutions are superimposed. Isocontours are plotted every second.

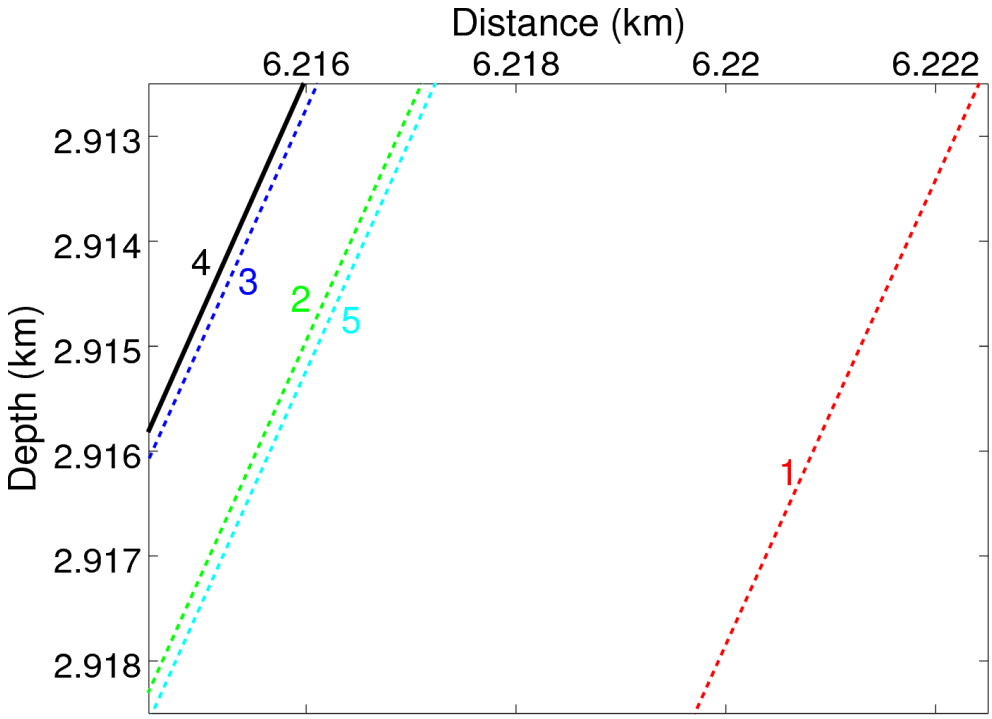


Figure 2.10: Isocontours at $t = 2$ s of the solutions for the homogeneous TTI medium. 1: DG computation with no factorization. 2: DG computation with elliptical reference solution. 3: DG computation with anelliptical reference solution. 4: Exact solution. 5: FD computation.

properly stored, I am able to proceed with the DG solver which calls these values when needed. This yields the highly accurate deep blue line in figure 2.10.

Finally, a finite-difference solution is computed inside the same medium using the iterative fast-sweeping factored TTI Eikonal solver detailed in Waheed et al. (2015b) and Tavakoli F. et al. (2015). For that purpose, a finite-difference grid composed of 277×1105 points is considered, so that the number of degrees of freedom is equivalent to the DG discretization: 306085. This yields the light blue line in figure 2.10.

Exact wavefronts in the whole medium are plotted in figure 2.9. A zoom on the isocontour corresponding to the time $t = 2$ s is shown in figure 2.10. There is an obvious improvement between the first computation with no factorization and the second computation which uses the factorization with an elliptical reference solution. The third result (deep blue line) is nearly mingled with the exact solution (black line). This illustrates the great advantage of using precomputed anelliptical values as the reference solution. Finally, the FD computation, with a comparable number of degrees of freedom, exhibits a solution which is more or less equivalent to the DG computation with the elliptical solution as reference for the factorization. These results illustrate the good behavior of the RK-DG solver regarding the TTI configuration. Factorization yields a great improvement of the solution, and I have shown that it is possible to use a highly accurate approximation of the anelliptical exact solution as the reference solution for the factorization.

2.4.2 Volcano structure with non-flat topography

Here, I illustrate the flexibility of the DG approach and its ability to handle non-flat topographies. I use a Gaussian topography simulating a volcanic dome. The TTI model is shown in figure 2.11. A magma chamber is mimicked by a high-speed zone located at the vertical under the dome. The speed reaches 4 km s^{-1} at its maximum, while it is 2 km s^{-1} away from the high-speed structure. Thomsen's parameters ϵ and δ take values respectively 0.4 and 0.2 away from the high-speed structure, while their values decrease inside the magma chamber, reaching 0 at their minimum. The tilt angle θ follows the topography, its absolute value decreases with depth until 0 at $z = 4$ km.

To compute the traveltime in this medium, I build a triangular mesh composed of 105442 P^1 triangular elements; a subset of the mesh is shown in figure 2.12.

Computations are performed for various source locations, at the surface as well as at depth. Isocontours are superimposed over the velocity model in figure 2.13. In figure 2.14, isocontours are superimposed over the maps of the spatial derivatives of the traveltime, for a source located at $x_s = 3.04$ km, $z_s = 0.63$ km. For this source location, a line of discontinuity of both derivatives is visible on these maps. This discontinuity matches with a singular part of the traveltime field, namely the kink at the junction between the direct wave on the right part and the refracted wave passing through the high-speed structure.

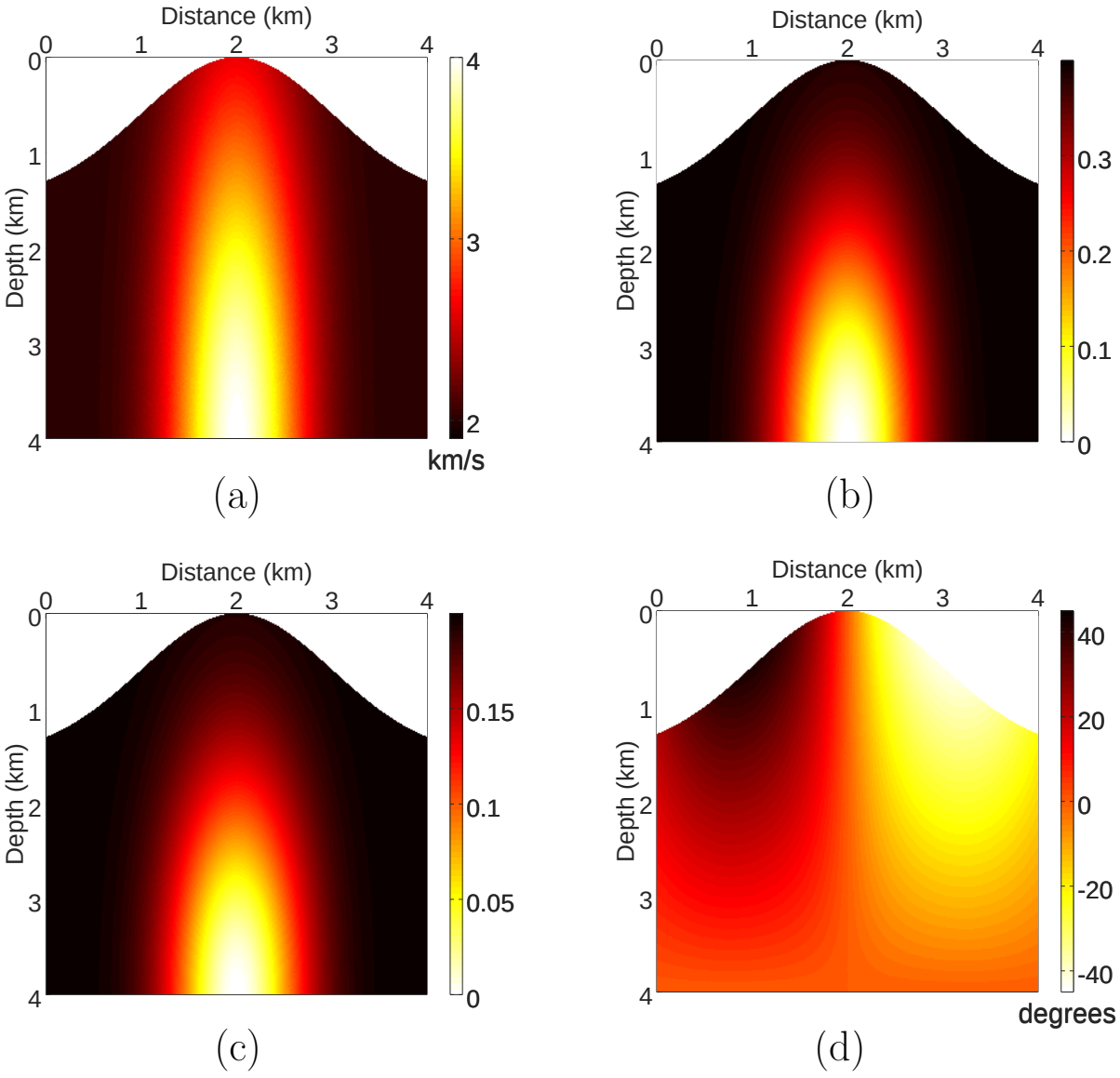


Figure 2.11: The volcano TTI model: (a) vertical velocity, (b) Thomsen's epsilon parameter, (c) Thomsen's delta parameter, (d) tilt angle.

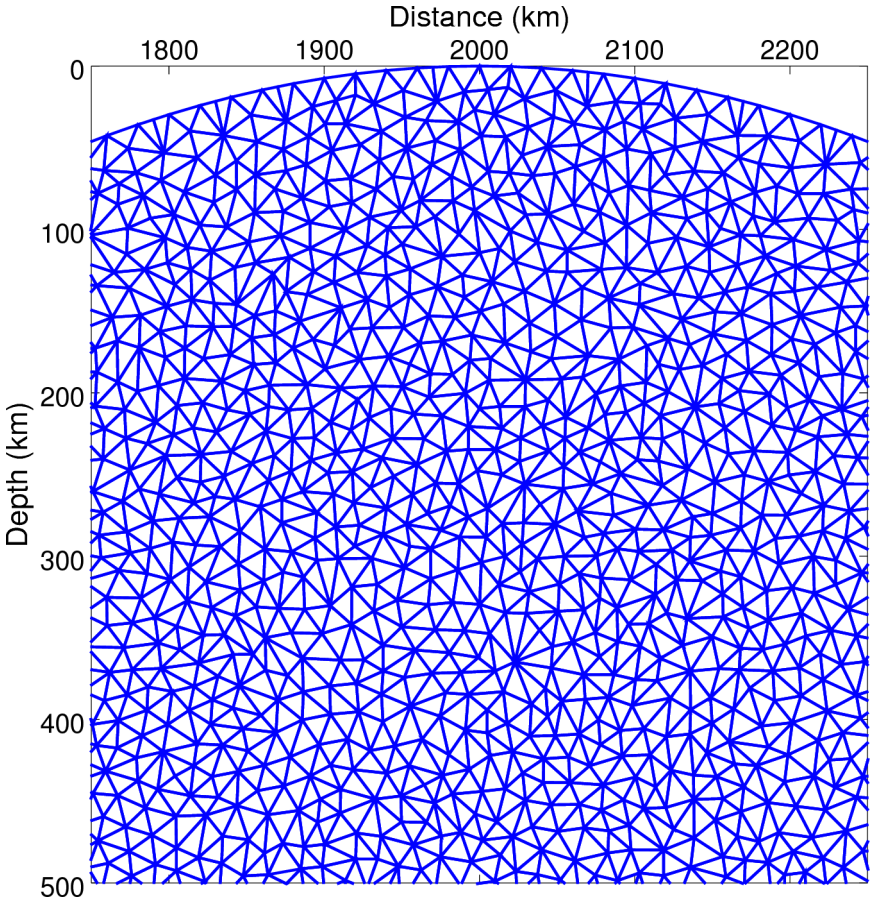


Figure 2.12: Detail of the triangular mesh built for the volcanic structure. The topography is sampled by edges of triangles.

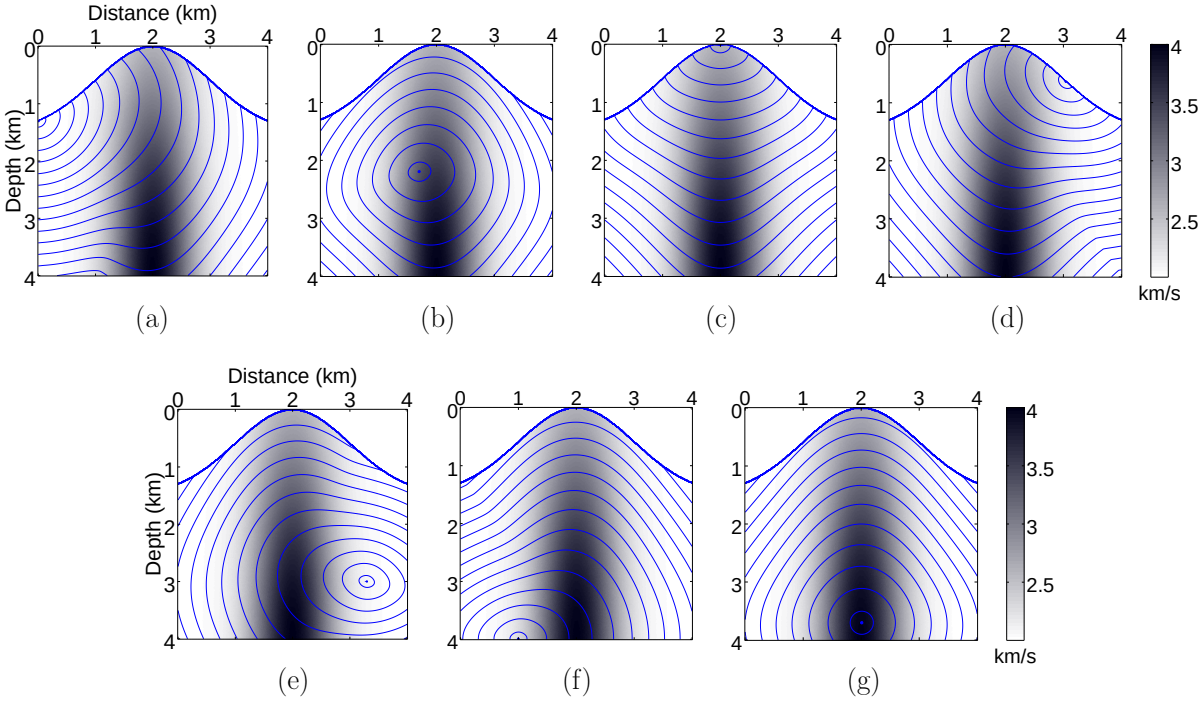


Figure 2.13: Isocontours computed in the volcano model for several source positions in km: (a) $x_s = 0$, $z_s = 1.3$, (b) $x_s = 1.7$, $z_s = 2.2$, (c) $x_s = 2$, $z_s = 0$, (d) $x_s = 3.04$, $z_s = 0.63$, (e) $x_s = 3.3$, $z_s = 3$, (f) $x_s = 1$, $z_s = 4$, (g) $x_s = 2$, $z_s = 3.7$. Isocontours are plotted every 0.1 second.

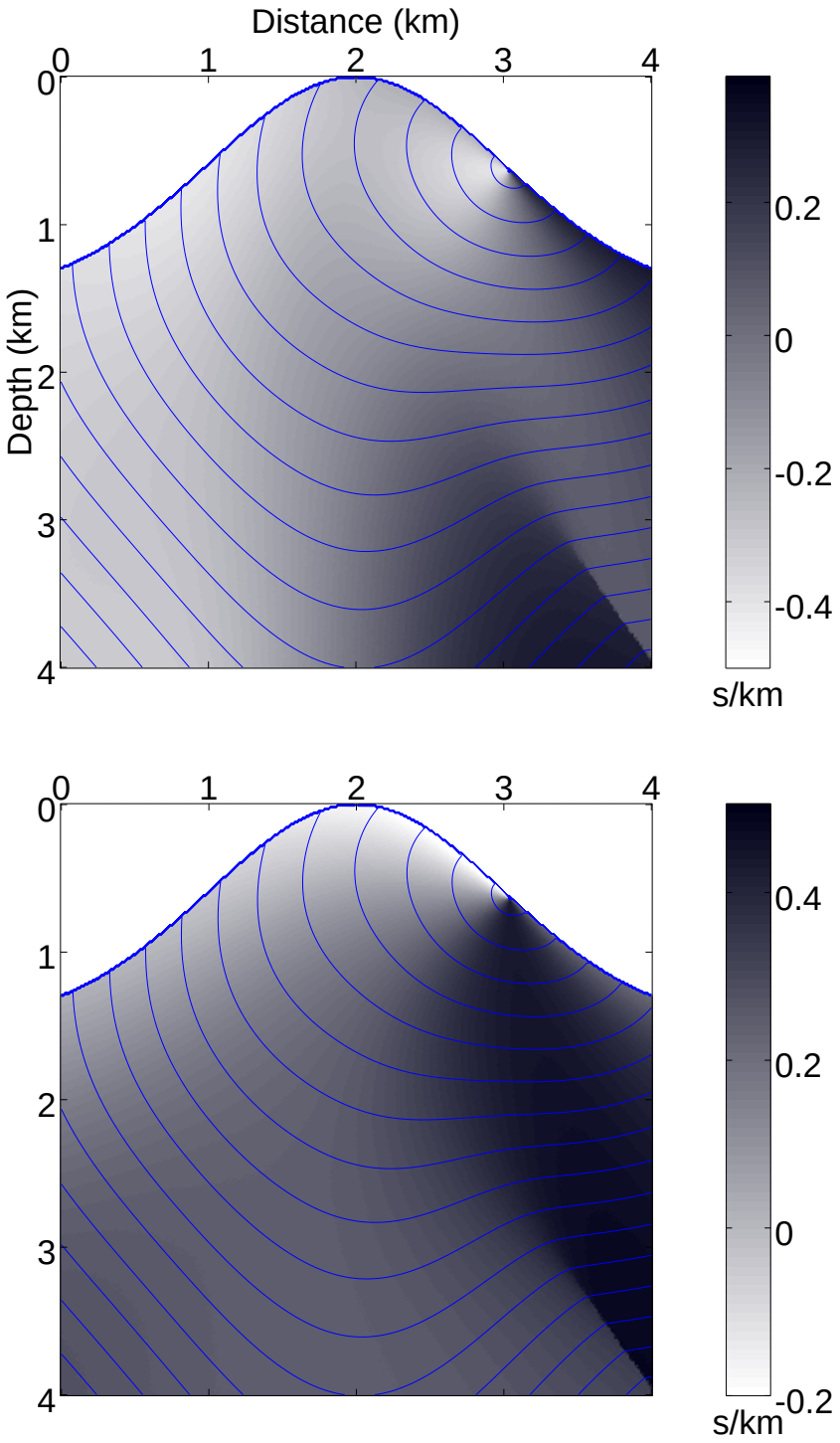


Figure 2.14: Maps of spatial derivatives of the traveltime and isocontours superimposed (blue lines), computed in the volcano model for a source point located at $x_s = 3.04$, $z_s = 0.63$ km. Top: x-derivative. Bottom: z-derivative. Isocontours are plotted every 0.1 second.

2.4.3 Realistic TTI model

In this last illustration, I test the RK-DG solver on the 2D BP TTI benchmark model from Shah (2007), which is used in the geophysics community for testing and validating modeling tools. The BP model is described by a highly contrasted P-wave velocity over a distance of 79 km and a depth of 11 km and corresponding Thomsen's parameters ϵ, δ and tilt angle θ shown in figure 2.15. In order to mitigate the impact of the discretization, the original model has been smoothed with Gaussian characteristic lengths of 200 m in both x and z directions. The DG solver proceeds over a Cartesian mesh of 179200 P^1 elements with $N_x = 1120$ and $N_z = 160$. The corresponding number of degrees of freedom is 537600. The same finite-difference solver as in the homogeneous TTI illustration is used for comparison purpose, which proceeds over a 1938×278 grid, so that the number of degrees of freedom is similar (538764). I proceed with various source locations, at the surface as well as in depth. Isocontours obtained with the two methods are shown in figure 2.16. In all the cases, they are very similar. The complexity of the medium yields complex propagation phenomena illustrated by the tortuous shape of the isocontours in some parts of the medium. Among many phenomena, a refracted wave is visible on top-left part of the first panel (source located at $x_s = 0$ km, z_s=0 km) due to the high-speed salt body at $x = 7$ km.

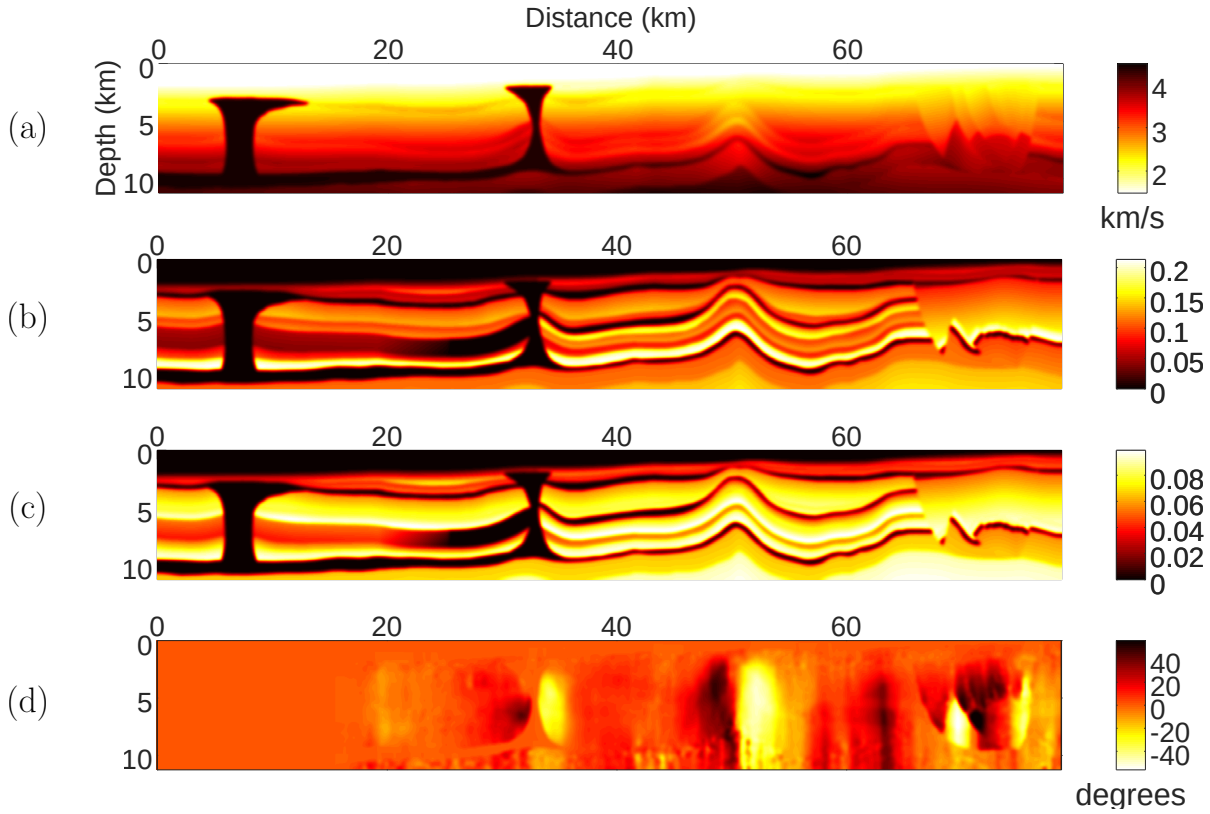


Figure 2.15: Smoothed 2D BP TTI model. From top to bottom: (a) vertical velocity, (b) Thomsen's epsilon parameter, (c) Thomsen's delta parameter, (d) tilt angle.

Isocontours are superimposed over the maps of the spatial derivatives of the traveltime in figure 2.17, for a source located at $x_s = 33$ km, $z_s = 3$ km. A detailed view in a smaller area is shown in figure 2.18. Although the traveltime field is continuous, discontinuities of the spatial derivatives of the

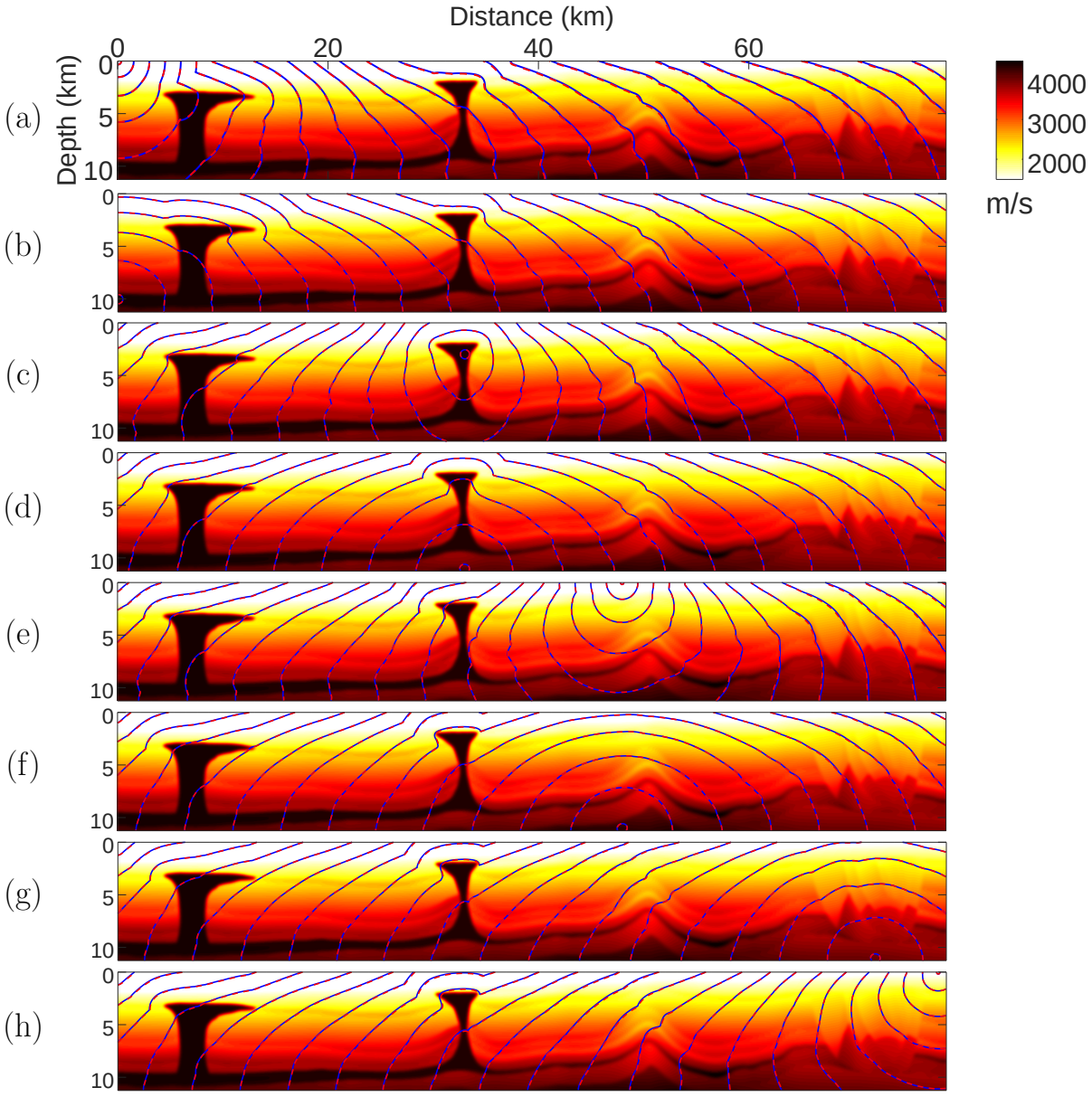


Figure 2.16: Isocontours computed in the smoothed BP TTI model for several source positions in km: (a) $x_s = 0$, $z_s = 0$, (b) $x_s = 0$, $z_s = 10$, (c) $x_s = 33$, $z_s = 3$, (d) $x_s = 33$, $z_s = 11$, (e) $x_s = 48$, $z_s = 0$, (f) $x_s = 48$, $z_s = 11$, (g) $x_s = 72$, $z_s = 11$, (h) $x_s = 78$, $z_s = 0$. Blue plain line: DG P^1 computation. Red dashed line: FD computation. Isocontours are plotted every second.

traveltime are prominent on these maps. They match with angles observed in the isocontours (singularities). The singularities of the solution are due to the viscosity solution which selects the lowest traveltime value where different phases compete. A line of discontinuity occurs where two branches of the solution meet (shock). Since the directions of propagation from both sides differ, the traveltime derivatives are discontinuous at these shocks. The resolution of a shock is related to the size of the element inside which it occurs. Outside of this element, the solution is not affected.

The same computation is performed using P^2 elements, with $N_x = 791$ and $N_z = 113$, yield-

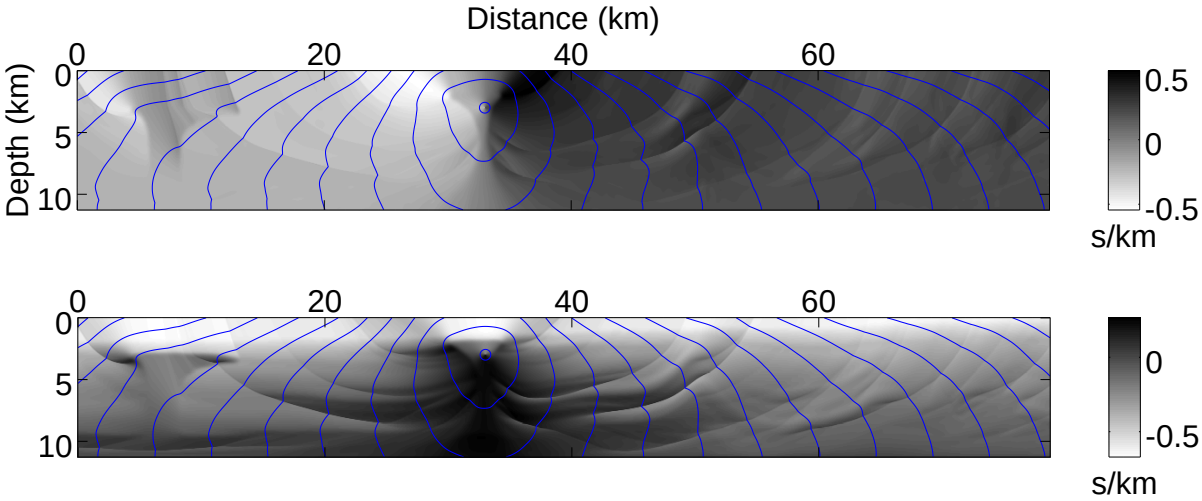


Figure 2.17: Maps of spatial derivatives of the traveltime and isocontours superimposed (blue line), computed in the smoothed BP TTI model for a source point located at $x_s = 33$, $z_s = 3$ km. P^1 computation with $N_x = 1120$ and $N_z = 160$. Top: x-derivative. Bottom: z-derivative. Isocontours are plotted every second.

ing 536298 degrees of freedom. Profiles of the traveltime and its derivatives along $x = 47$ km and $z = 6.7$ km are shown in figure 2.19. In figure 2.20, the profiles along $x = 47$ km of the traveltime derivatives computed with P^1 and P^2 are compared near the shock occurring at 6.8 km. P^1 yields a piecewise constant approximation of the derivatives, whereas P^2 yields a piecewise linear one. Regarding the derivatives, the shock is poorly approximated inside the element where it occurs, but this does not affect the solution elsewhere. For practical applications, one has to be careful if the solution inside such an element is needed. A criterion based on the variations of the derivatives of the solution could be designed in order to refine the mesh and recompute locally the solution with a better resolution of the shock.

In a finite-difference strategy, the traveltime field, as well as its derivatives, are computed at grid points, then interpolated if needed somewhere else. However, in the DG approach, these quantities are directly accessed at any point inside each element to a given order of accuracy, from the polynomial approximation. I emphasize that no interpolation is required to obtain the maps of figure 2.17 nor the profiles of figures 2.19 and 2.20.

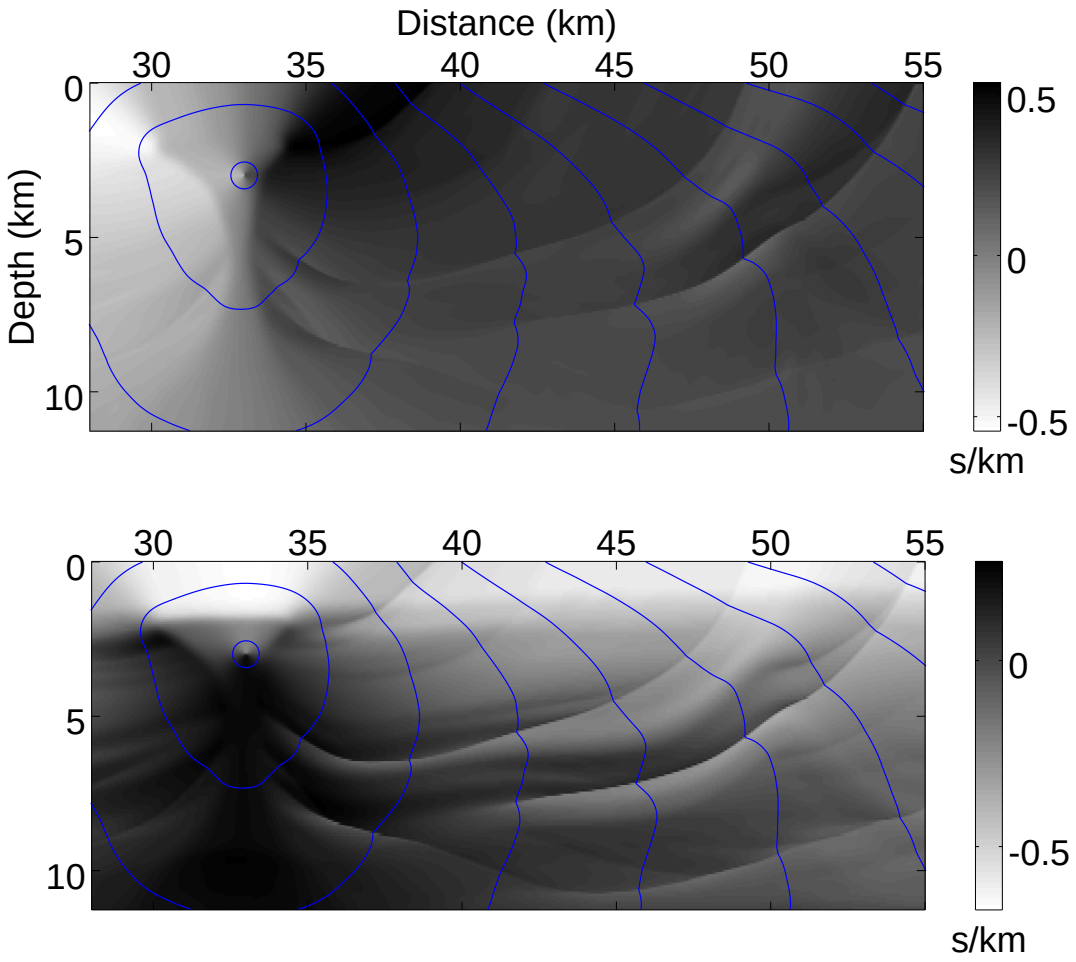


Figure 2.18: Maps of spatial derivatives of the traveltime and isocontours superimposed (blue line), computed in the smoothed BP TTI model for a source point located at $x_s = 33$, $z_s = 3$ km. P^1 computation with $N_x = 1120$ and $N_z = 160$. Zoom in the [28, 55]×[0, 11] rectangle. Top: x-derivative. Bottom: z-derivative. Isocontours are plotted every second.

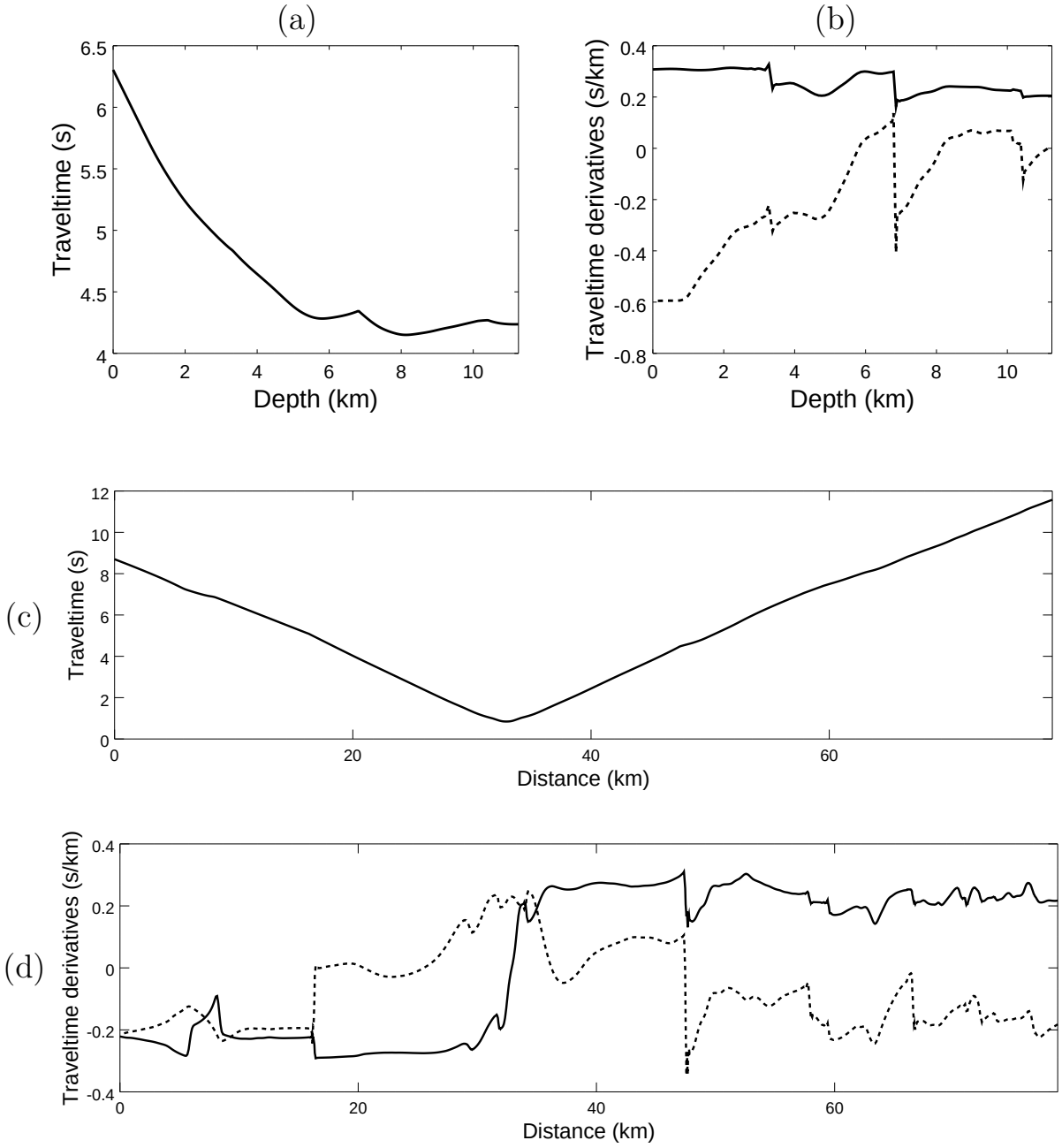


Figure 2.19: Profiles computed in the smoothed BP TTI model for a source point located at $x_s = 33$, $z_s = 3$ km. P^2 computation with $N_x = 791$ and $N_z = 113$. Profile along $x = 47$ km: (a) Traveltime field; (b) x-derivative (plain line) and z-derivative (dashed line) of the traveltime. Profile along $z = 6.7$ km: (c) Traveltime field; (d) x-derivative (plain line) and z-derivative (dashed line) of the traveltime. These plots highlight the smoothness of the traveltime field compared to its discontinuous derivatives.

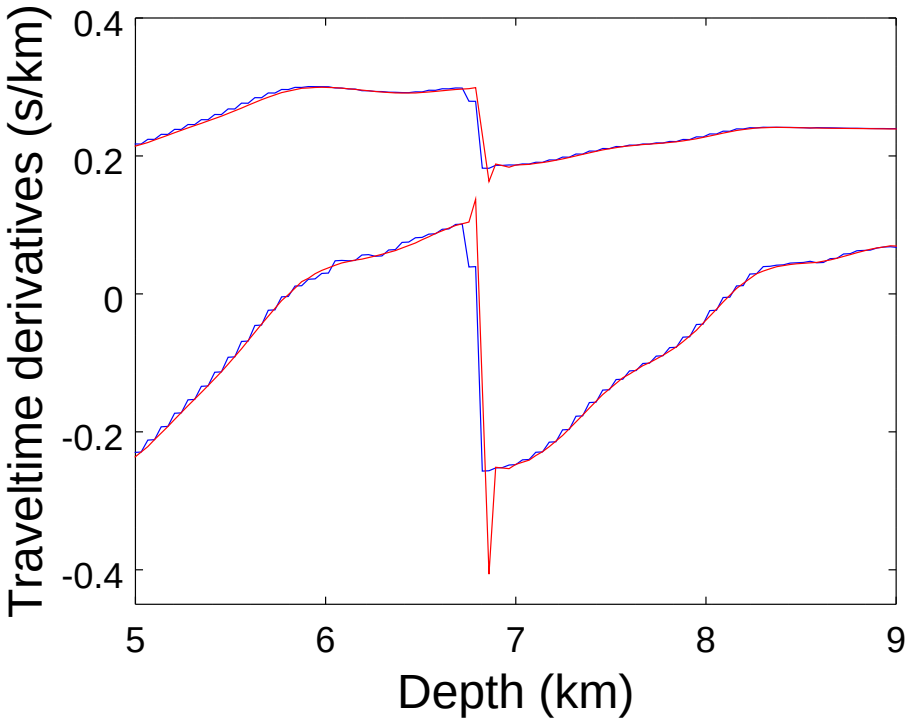


Figure 2.20: Profile of the traveltime derivatives along $x = 47$ km computed in the smoothed BP TTI model for a source point located at $x_s = 33$, $z_s = 3$ km. P^1 computation (blue line) and P^2 computation (red line) with similar numbers of degrees of freedom. Top lines: x-derivative; bottom lines: z-derivative. P^1 yields a piecewise constant approximation of the derivatives, whereas P^2 yields a piecewise linear one. Note the local variation of the viscous solution quite sensitive to the element size and the polynomial interpolation while the solution accuracy is not impacted elsewhere.

2.5 Conclusion

In this chapter, I have presented an RK-DG method for solving time-dependent Hamilton–Jacobi equations with a point-source condition. The steady state is the viscosity solution to the corresponding static Eikonal equation, namely first-arrival traveltime. This high-order accurate, compact and flexible method computes traveltime and its spatial derivatives in heterogeneous anisotropic media. I emphasize that scheme (2.39) is written in a general Hamiltonian formulation, thus it may hold for a large variety of Hamiltonians, opening doors to other types of anisotropy and applications. Moreover, the local computation for the RK steps is explicit, thus it does not require to find the roots of local polynomial equations.

Although not computationally efficient for steady-state computation, as I will explain in the next chapter, it is worth noticing that the RK-DG solver has a great interest for time-dependent problems, since it is efficient and accurate. In the geophysical context, it could be employed to solve the multi-valued traveltime problem in a paraxial approach, where the depth is taken as the evolution parameter (pseudo-time) and the traveltime is computed step by step in depth, assuming downward propagating rays (see e.g. Qian and Symes, 2001). More details are given at the end of this manuscript.

Appendices

2.A Equivalence of VTI Hamiltonians

In section 2.2.2.1, in the 2D VTI case, I introduce Hamiltonian (2.38) to replace Hamiltonian (1.44). Those two Hamiltonians are expected to yield the same steady state. This is what I demonstrate in this appendix. At steady state, with the notations $X = u_{,x}, Z = u_{,z}$, inserting (1.44) inside (2.1) writes

$$dX^2 + eZ^2 + cX^2Z^2 - 1 = 0. \tag{2.49}$$

Adding a term on both sides and rearranging (2.49) yields

$$1 + \frac{1}{4}\left(dX^2 + eZ^2\right)^2 - dX^2 - eZ^2 = \frac{1}{4}\left(dX^2 + eZ^2\right)^2 + cX^2Z^2, \tag{2.50}$$

$$\left[1 - \frac{1}{2}\left(dX^2 + eZ^2\right)\right]^2 = \frac{d^2}{4}X^4 + \frac{e^2}{4}Z^4 + \left(\frac{de}{2} + c\right)X^2Z^2. \tag{2.51}$$

From (1.42) I get

$$\frac{de}{2} + c = \frac{V_P^4}{2}\left(1 - 2\epsilon + 4\delta\right), \tag{2.52}$$

which is positive in practice for all realistic applications in geophysics. Therefore, the square root of (2.51) can be taken without loss of generality, yielding

$$\sqrt{\frac{1}{4}\left(dX^2 + eZ^2\right)^2 + cX^2Z^2} + \frac{1}{2}\left(dX^2 + eZ^2\right) = 1. \tag{2.53}$$

Since $d > 0$ and $e > 0$, the square root can be taken again, and dividing by $\sqrt{d}$, the VTI Hamiltonian (2.38) is obtained:

$$\mathcal{H}_{VTI} = \frac{1}{\sqrt{d}}\left(\sqrt{\sqrt{\frac{1}{4}\left(du_{,x}^2 + eu_{,z}^2\right)^2 + cu_{,x}^2u_{,z}^2} + \frac{1}{2}\left(du_{,x}^2 + eu_{,z}^2\right)} - 1\right). \tag{2.54}$$

2.B Lipschitz continuity of the VTI Hamiltonian

In this appendix, I demonstrate the Lipschitz continuity of the 2D VTI Hamiltonian (2.38) introduced in section 2.2.2.1. With the notations $X = u_{,x}, Z = u_{,z}$, for $(X, Z) \neq (0, 0)$, the derivatives of VTI Hamiltonian (2.38) write

$$\mathcal{H}_1(X,Z) = \frac{1}{\sqrt{d}} \frac{dX + \frac{2cXZ^2 + dX(dX^2+eZ^2)}{\sqrt{(dX^2+eZ^2)^2+4cX^2Z^2}}}{\sqrt{2\sqrt{\left(dX^2 + eZ^2\right)^2 + 4cX^2Z^2} + 2\left(dX^2 + eZ^2\right)}}, \tag{2.55}$$

$$\mathcal{H}_2(X,Z) = \frac{1}{\sqrt{d}} \frac{eZ + \frac{2cX^2Z + eZ(dX^2+eZ^2)}{\sqrt{(dX^2+eZ^2)^2+4cX^2Z^2}}}{\sqrt{2\sqrt{\left(dX^2 + eZ^2\right)^2 + 4cX^2Z^2} + 2\left(dX^2 + eZ^2\right)}}. \tag{2.56}$$

This yields

$$\|\boldsymbol{\mathcal{H}}\|^2 = \frac{1}{2d}\frac{\left(dX + \frac{2cXZ^2+dX(dX^2+eZ^2)}{\sqrt{(dX^2+eZ^2)^2+4cX^2Z^2}}\right)^2 + \left(eZ + \frac{2cX^2Z+eZ(dX^2+eZ^2)}{\sqrt{(dX^2+eZ^2)^2+4cX^2Z^2}}\right)^2}{\sqrt{\left(dX^2+eZ^2\right)^2+4cX^2Z^2}+dX^2+eZ^2}. \tag{2.57}$$

In the elliptical case, $c = 0$ thus (2.57) simplifies:

$$\|\boldsymbol{\mathcal{H}}\|^2_{\text{ELL}} = \frac{d^2X^2+e^2Z^2}{d^2X^2+deZ^2}. \tag{2.58}$$

Since $\epsilon \geqslant 0$, then $d > e$ and we have

$$\max \|\boldsymbol{\mathcal{H}}\|_{\text{ELL}} = 1, \tag{2.59}$$

which is obtained for $Z = 0$.

In the general anelliptical case, I use the polar coordinate system

$$\begin{aligned} \sqrt{d}X &= r\cos\gamma, \\ \sqrt{e}Z &= r\sin\gamma. \end{aligned} \tag{2.60}$$

Changing variables in (2.57) yields

$$\|\boldsymbol{\mathcal{H}}\|^2 = \frac{A+B+C}{2d\left(1+\sqrt{1+\frac{4c}{de}\cos^2\gamma\sin^2\gamma}\right)}, \tag{2.61}$$

with

$$\begin{aligned} A &= d\cos^2\gamma + e\sin^2\gamma, \\ B &= \frac{d\cos^2\gamma + e\sin^2\gamma + 4c\cos^2\gamma\sin^2\gamma(\frac{1}{d}+\frac{1}{e}) + \frac{4c^2}{de}\cos^2\gamma\sin^2\gamma(\frac{\cos^2\gamma}{d}+\frac{sin^2\gamma}{e})}{1+\frac{4c}{de}\cos^2\gamma\sin^2\gamma}, \\ C &= \frac{2(d\cos^2\gamma + e\sin^2\gamma) + 4c\cos^2\gamma\sin^2\gamma(\frac{1}{d}+\frac{1}{e})}{\sqrt{1+\frac{4c}{de}\cos^2\gamma\sin^2\gamma}}. \end{aligned} \tag{2.62}$$

Expression (2.61) holds for $r \neq 0$. The variable r simplifies so that $\|\boldsymbol{\mathcal{H}}\|^2$ is a function of one variable γ. This function is π-periodic.

Getting back to the Thomsen's parameters ϵ and δ using (1.42), some calculus that I do not reproduce here gives

$$\|\boldsymbol{\mathcal{H}}\|^2 = \frac{D+E}{(2+4\epsilon)\left(1-\frac{2(\epsilon-\delta)}{1+2\epsilon}\sin^2 2\gamma\right)\left(1+\sqrt{1-\frac{2(\epsilon-\delta)}{1+2\epsilon}\sin^2 2\gamma}\right)} + 1, \tag{2.63}$$

with

$$\begin{aligned} D &= \frac{4\epsilon(\epsilon-\delta)(1+4\epsilon-2\delta)}{(1+2\epsilon)^2}\sin^2\gamma\sin^2 2\gamma + \frac{4(\epsilon-\delta)(\epsilon(1+2\epsilon)+\epsilon-\delta)}{(1+2\epsilon)^2}\sin^2 2\gamma - (6\epsilon-2\delta)\sin^2\gamma, \\ E &= 4\epsilon\sqrt{1-\frac{2(\epsilon-\delta)}{1+2\epsilon}\sin^2 2\gamma}\left(\frac{\epsilon-\delta}{1+2\epsilon}\sin^2 2\gamma - \sin^2\gamma\right). \end{aligned} \tag{2.64}$$

Since $\epsilon \geqslant \delta \geqslant 0$ and assuming (2.52) is positive, we have

$$\begin{aligned} \frac{\epsilon-\delta}{1+2\epsilon}\sin^2 2\gamma - \sin^2\gamma &= \sin^2\gamma\left(\frac{4(\epsilon-\delta)}{1+2\epsilon}(1-\sin^2\gamma)-1\right) \\ &\leqslant \frac{4(\epsilon-\delta)}{1+2\epsilon}-1 \leqslant 0. \end{aligned} \tag{2.65}$$

Therefore, $E \leqslant 0$.

To state $D \leqslant 0$, I define $y = \sin^2 2\gamma$. Thus

$$\begin{aligned} D &= y\left(4A'y(1-y)+4B'(1-y)-C'\right) \\ &= y\left(-4A'y^2+(4A'-4B')y+4B'-C'\right), \end{aligned} \tag{2.66}$$

with the notations

$$\begin{aligned} A' &= \frac{4\epsilon(\epsilon-\delta)(1+4\epsilon-2\delta)}{(1+2\epsilon)^2}, \\ B' &= \frac{4(\epsilon-\delta)(\epsilon(1+2\epsilon)+\epsilon-\delta)}{(1+2\epsilon)^2}, \\ C' &= 6\epsilon-2\delta. \end{aligned} \tag{2.67}$$

Since $y \geqslant 0$, D is of the sign of the second-order polynomial in y in (2.66). Its discriminant writes

$$\Delta = 16\left((A'-B')^2+A'(4B'-C')\right) = 16\left((A'+B')^2-A'C'\right). \tag{2.68}$$

Using (2.67), we obtain

$$\begin{aligned} (A'+B')^2-A'C' &= \frac{4(\epsilon-\delta)(3\epsilon-\delta)}{(1+2\epsilon)^2}\big(4(\epsilon-\delta)(3\epsilon-\delta)-2\epsilon(1+4\epsilon-2\delta)\big) \\ &= \frac{4(\epsilon-\delta)(3\epsilon-\delta)}{(1+2\epsilon)^2}\big(2\epsilon(2\epsilon-4\delta-1)-4\delta(\epsilon-\delta)\big) \\ &\leqslant 0, \end{aligned} \tag{2.69}$$

since (2.52) is positive. It follows that $D \leqslant 0$.

2.C Integral computation in 2D

2.C.1 Reference cells and mappings

The triangular case The integral computations in scheme (2.39) are performed using a reference cell. Details are given here for triangular meshes. Consider the mapping

$$\begin{array}{rccc} \gamma_i & : & \hat{K} & \longrightarrow & K_i \\ & & (\xi,\eta) & \longmapsto & (x_i,z_i), \end{array} \tag{2.70}$$

with

$$\begin{aligned} x_i &= x_i^1+(x_i^2-x_i^1)\xi+(x_i^3-x_i^1)\eta, \\ z_i &= z_i^1+(z_i^2-z_i^1)\xi+(z_i^3-z_i^1)\eta, \end{aligned} \tag{2.71}$$

where $(x_i^j, z_i^j)_{1\leqslant j\leqslant 3}$ are the vertex of K_i. The reference triangle is shown in figure 2.21. The inverse mapping of γ_i writes

$$\begin{array}{rccc} \gamma_i^{-1} & : & K_i & \longrightarrow & \hat{K} \\ & & (x_i, z_i) & \longmapsto & (\xi, \eta), \end{array} \tag{2.72}$$

with

$$\begin{aligned} \xi &= \frac{1}{|J_i|}\left((z_i^3 - z_i^1)x + (x_i^1 - x_i^3)z + x_i^3 z_i^1 - x_i^1 z_i^3\right), \\ \eta &= \frac{1}{|J_i|}\left((z_i^1 - z_i^2)x + (x_i^2 - x_i^1)z + x_i^1 z_i^2 - x_i^2 z_i^1\right), \\ |J_i| &= (x_i^2 - x_i^1)(z_i^3 - z_i^1) - (x_i^1 - x_i^3)(z_i^1 - z_i^2) = 2\Delta K_i. \end{aligned} \tag{2.73}$$

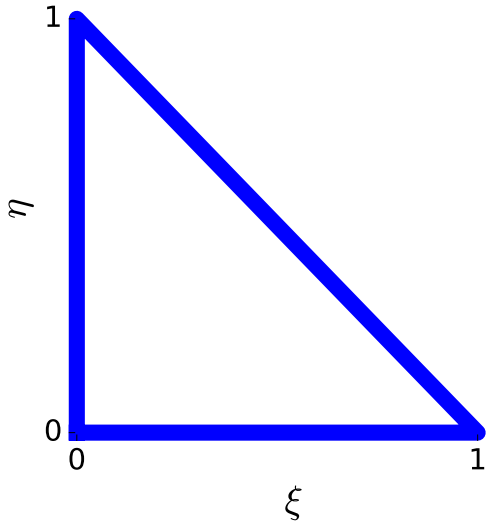


Figure 2.21: Reference triangle for the DG method. Vertex #1 with coordinates (x_i^1, z_i^1) of a current element K_i maps to the bottom-left vertex of the reference element with coordinates $(\xi, \eta) = (0, 0)$. Similarly, vertex #2 maps to $(1, 0)$, and vertex #3 maps to $(0, 1)$.

For $1 \leqslant j \leqslant d$ with d being the number of degrees of freedom, let $\hat{\phi}^j = \phi_i^j \circ \gamma_i$ a basis function defined over $\hat{K}$. We are able to compute integrals which involve the numerical solution writing like (2.25) by using only reference basis functions that are defined in the reference space and making a change of variables. For example, we have

$$\int_{K_i} \phi_i^j(\boldsymbol{x})\phi_i^k(\boldsymbol{x})\mathrm{d}\boldsymbol{x} = |J_i| \int_{\hat{K}} \hat{\phi}^j(\xi, \eta)\hat{\phi}^k(\xi, \eta)\mathrm{d}\xi\mathrm{d}\eta. \tag{2.74}$$

This is also true for spatial derivatives of the numerical solution. Using the chain rule, we have

$$\nabla\phi_i^j(\boldsymbol{x}) = D\left(\gamma_i^{-1}(x_i, z_i)\right)^T \nabla_{(\xi,\eta)}\hat{\phi}^j(\xi, \eta), \tag{2.75}$$

where

$$D\left(\gamma_i^{-1}(x_i, z_i)\right) = \frac{1}{|J_i|}\begin{pmatrix} z_i^3 - z_i^1 & x_i^1 - x_i^3 \\ z_i^1 - z_i^2 & x_i^2 - x_i^1 \end{pmatrix}. \tag{2.76}$$

The rectangular case For a rectangular Cartesian mesh, the mapping γ_i of an element K_i to the reference square is simple. We have

$$\begin{aligned} x_i &= x_i^1 + (x_i^2 - x_i^1)\xi, \\ z_i &= z_i^1 + (z_i^3 - z_i^1)\eta. \end{aligned} \tag{2.77}$$

The Jacobian derivation and the computation of integrals are then straightforward.

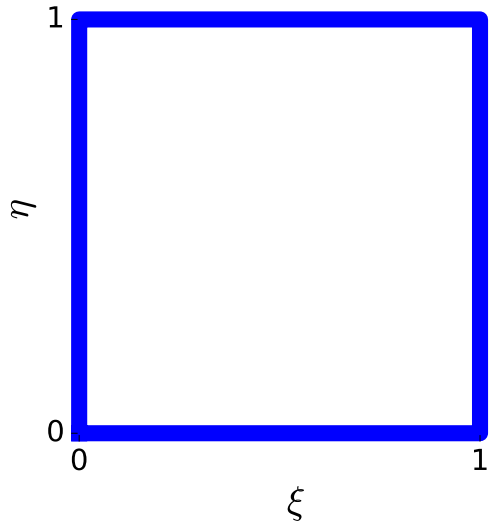


Figure 2.22: Reference rectangle for the DG method. Vertex #1 with coordinates (x_i^1, z_i^1) of a current element K_i maps to the bottom-left vertex of the reference element with coordinates $(\xi, \eta) = (0, 0)$. Similarly, vertex #2 maps to $(1, 0)$, and vertex #3 maps to $(0, 1)$. The fourth vertex is the opposite corner to vertex #1, and its coordinates do not appear in the mapping because we consider a rectangle.

In conclusion, the integral computations are performed in the reference cell. Next, I explain how the numerical computation is performed in practice with quadrature rules and Gauss points.

2.C.2 Quadrature rules and Gauss points

Inside the reference element, the numerical estimations of edge and surface integrals rely on quadrature rules like

$$\begin{aligned} \int_0^1 f(s)\mathrm{d}s &\simeq \sum_{k=1}^{G_1} f(s_1^k) w_1^k, \\ \int_0^1 \int_0^1 f(\xi, \eta)\mathrm{d}\xi\mathrm{d}\eta &\simeq \sum_{k=1}^{G_2} f(\xi_2^k, \eta_2^k) w_2^k. \end{aligned} \tag{2.78}$$

The integrals are replaced by a sum of weighted integrands evaluated at specific locations, namely Gauss points. In expressions (2.78), the subscripts 1 and 2 denote Gauss points coordinates and associated weights for one-dimension and two-dimension integrals, respectively. The number of Gauss points G_1 for edge integrals and G_2 for surface integrals required for optimal accuracy depend on the

approximation order of the elements. For instance, for the P^1 rectangular case, I use

$$\begin{aligned} &G_1 = 2, \\ &s_1^1 = \frac{1}{2} - \frac{1}{2\sqrt{3}}, \quad s_1^2 = \frac{1}{2} + \frac{1}{2\sqrt{3}}, \\ &w_1^1 = w_1^2 = \frac{1}{2}, \end{aligned} \tag{2.79}$$

and

$$\begin{aligned} &G_2 = 4, \\ &\xi_2^1 = \frac{1}{2} - \frac{1}{2\sqrt{3}}, \quad \eta_2^1 = \frac{1}{2} - \frac{1}{2\sqrt{3}}, \\ &\xi_2^2 = \frac{1}{2} + \frac{1}{2\sqrt{3}}, \quad \eta_2^2 = \frac{1}{2} - \frac{1}{2\sqrt{3}}, \\ &\xi_2^3 = \frac{1}{2} - \frac{1}{2\sqrt{3}}, \quad \eta_2^3 = \frac{1}{2} + \frac{1}{2\sqrt{3}}, \\ &\xi_2^4 = \frac{1}{2} + \frac{1}{2\sqrt{3}}, \quad \eta_2^4 = \frac{1}{2} + \frac{1}{2\sqrt{3}}, \\ &w_2^1 = w_2^2 = w_2^3 = w_2^4 = \frac{1}{4}. \end{aligned} \tag{2.80}$$

2.D Basis functions in 2D geometry

I present here the basis functions that I use in 2D geometry, either in the triangular or in the rectangular case. Indeed, the mathematical properties of the DG method are not affected by the particular choice of basis functions, but only by the local approximation spaces $\mathcal{P}_i$. However, the choice of the basis functions has an influence on the computational cost. In my work, I do not use a nodal approach, which rely on Lagrange polynomials based on a set of nodes. Instead, I use modal basis functions, as proposed in Piperno (2005). They yield an easy computation of the mass matrix and its inverse, and an efficient computation of edge integrals. I have not compared the performance of this modal set with nodal basis functions, though.

Triangular case Inside the reference triangle, three fundamental basis functions are defined as

$$\begin{aligned} &\phi^1(\xi, \eta) = \xi, \\ &\phi^2(\xi, \eta) = \eta, \\ &\phi^3(\xi, \eta) = 1 - \xi - \eta. \end{aligned} \tag{2.81}$$

These basis functions actually give the barycentric coordinates of any point of coordinates (ξ, η) with respect to the three vertices of the triangle.

A P^1 element employs these three basis functions to span the space of first-order polynomials and to describe the numerical solution in this space. Inside this element, the solution writes

$$u(\xi, \eta) = (a - c)\xi + (b - c)\eta + c, \tag{2.82}$$

where a, b, and c are the degrees of freedom associated with the three basis functions defined in (2.81), respectively. The basis functions for a P^2 element write

$$\begin{aligned}
\phi^1(\xi,\eta) &= \xi^2,\\
\phi^2(\xi,\eta) &= 2\xi\eta,\\
\phi^3(\xi,\eta) &= \eta^2,\\
\phi^4(\xi,\eta) &= 2\xi(1-\xi-\eta),\\
\phi^5(\xi,\eta) &= 2\eta(1-\xi-\eta),\\
\phi^6(\xi,\eta) &= (1-\xi-\eta)^2.
\end{aligned} \tag{2.83}$$

A general formula is given in Piperno (2005) for any order of approximation. The author also gives mass matrices and their inverse.

Rectangular case Inside the reference rectangle, a natural formulation for the rectangular case would involve four fundamental functions, expressed as

$$\begin{aligned}
\phi^1(\xi,\eta) &= \xi\eta,\\
\phi^2(\xi,\eta) &= (1-\xi)\eta,\\
\phi^3(\xi,\eta) &= \xi(1-\eta),\\
\phi^4(\xi,\eta) &= (1-\xi)(1-\eta).
\end{aligned} \tag{2.84}$$

The associated P^1 approximation spans a space of solutions under the general form

$$u(\xi,\eta) = (a-b-c+d)\xi\eta + (c-d)\xi + (b-d)\eta + d, \tag{2.85}$$

where a, b, c, and d are the degrees of freedom associated with the four basis functions, respectively. The local solution is no longer a P^1 polynomial. An additional cross term has been added.

However, in practice, I do not consider the cross term, keeping only three degrees of freedom for an element as for the triangular case. The basis functions are the same as for triangles, but the mass matrices are different since the integrals are computed over the reference square instead of the triangle. This choice is motivated by numerical experiments: I have carefully tested both implementations and I verified that the convergence orders computed in the convergence analysis in section 2.3 are the same in both cases. Moreover, taking three fundamental functions instead of four yields a quite significant decrease of the computational cost.

2.E Wavefront curves in a 2D homogeneous VTI medium

I present here the parametric formulation giving the wavefront position at a given time t, and thus the group angle, for a given phase angle, in a 2D homogeneous anelliptical TI medium. The following formulas come from Carcione et al. (1988, eq. 5.9).

First, we define the quantities

$$\begin{aligned}
\alpha &= \frac{c_{33}}{c_{44}}, \\
\beta &= \frac{c_{11}}{c_{44}}, \\
\gamma &= 1 + \alpha\beta - \left(\frac{c_{13}}{c_{44}} + 1\right)^2, \\
k_1 &= 2\alpha(\beta+1) + \gamma(\alpha+1), \\
k_2 &= 2\beta(\alpha+1) - \gamma(\beta+1),
\end{aligned} \tag{2.86}$$

where c_{ij} are the components of the elasticity matrix of the homogeneous medium, using the Voigt notation.

For a phase angle φ, we then define

$$\begin{aligned}
A(\varphi) &= \alpha\cos^4\varphi + \gamma\cos^2\varphi\sin^2\varphi + \beta\sin^4\varphi, \\
B(\varphi) &= (\alpha+1)\cos^2\varphi + (\beta+1)\sin^2\varphi, \\
R(\varphi) &= \left[\frac{B(\varphi) - \left(B^2(\varphi) - 4A(\varphi)\right)^{1/2}}{2A(\varphi)}\right]^{1/2}.
\end{aligned} \tag{2.87}$$

The phase velocity $V(\varphi)$ then writes

$$V(\varphi) = \frac{1}{R(\varphi)}\left(\frac{c_{44}}{\rho}\right)^{1/2}, \tag{2.88}$$

and we finally obtain the (x, z) coordinates of the position of the ray at time t for a phase angle φ with the expressions

$$\begin{aligned}
x &= \frac{Vt\sin\varphi}{2A}\left[2\beta\sin^2\varphi + \gamma\cos^2\varphi + \frac{\cos^2\varphi(k_1\cos^2\varphi - k_2\sin^2\varphi)}{(B^2-4A)^{1/2}}\right], \\
z &= \frac{Vt\cos\varphi}{2A}\left[2\alpha\cos^2\varphi + \gamma\sin^2\varphi - \frac{\sin^2\varphi(k_1\cos^2\varphi - k_2\sin^2\varphi)}{(B^2-4A)^{1/2}}\right].
\end{aligned} \tag{2.89}$$

It is then straigthforward to compute the group angle ψ from the (x, z) coordinates (see figure 2.8).

Chapter 3

Solving 3D Eikonal equations with a fast-sweeping-DG method for accurate and efficient computation of traveltime

Contents

Introduction

The main focus of this chapter is on computational efficiency. I present a high performance implementation for the solution of 3D Hamilton–Jacobi equations, improving upon the DG scheme presented in chapter 2.

RK-DG practical limitation Despite the high accuracy provided by the DG discretization, the Runge–Kutta integration is known to be non-efficient for a steady-state computation. At each pseudo-time iteration, every cell of the mesh is updated, while only a small subset of cells is concerned by the front propagation. In terms of numerical complexity, in the RK-DG algorithm, if the typical length of the

mesh is divided by 2, the number of degrees of freedom is multiplied by 4, and the number of pseudo-time steps required to reach the steady state is multiplied by 2 due to the CFL condition. In theory, the computational complexity is thus $\mathcal{O}(N^3/2)$, where N is the number of degrees of freedom. This is what I observe in practice. To give a comparison in a 2D realistic case, the RK-DG computation for one source in the BP TTI model in the illustration of section 2.4.3 takes about fifty minutes on a 2.6 GHz machine with 8 GB of RAM. The equivalent FSM-FD computation on the same machine takes 20 seconds with the same number of degrees of freedom. The RK-DG prototype code could certainly be optimized. However, the linear complexity in $\mathcal{O}(N)$ of the fast-sweeping-based FD code makes it far more efficient. It is therefore desirable to try and integrate the efficient FSM approach in the accurate DG formulation, as an acceleration tool for reaching the steady state.

State of the art I have reviewed some FSM methods for solving static Eikonal equations in section 1.2.3 and some extensions to anisotropy in section 1.3.2. Most of these extensions consider 2D problems only, except in Waheed et al. (2015b) where 3D tilted orthorhombic (TOR) media are considered. Moreover, all the developments are performed using FD schemes, generally of first-order, or higher-order at high cost with non-compact stencils. The idea of integrating a Runge–Kutta time integration of the time-dependent Eikonal equation as a local solver into a global sweeping procedure has been tested in an FD framework by Zhang et al. (2005c). FSM is used as an acceleration loop over the time-marching procedure. In some sense, the block-Jacobi method is replaced by a block-Gauss–Seidel method, which exploits the directions of propagation of the solution by a suitable ordering of the elements. The elements of the mesh are updated sequentially instead of all together. Regarding DG approaches, a numerical strategy was developed in 2D to solve Hamilton–Jacobi equations with an FSM procedure over a DG discretization by Li et al. (2008) and Zhang et al. (2011). Based on the DG solver of Cheng and Shu (2007), this strategy exhibits some practical limitations. An initial guess is needed and deduced from a preliminary FD computation. The local scheme requires a cumbersome ℓ^2 reconstruction of the solution's derivatives at the cell interface. Moreover, this local scheme considers only piecewise-linear Cartesian cases. In addition, it is developed only for isotropic media. Finally, the point source singularity is not cured.

A new FSM-DG algorithm In this chapter, I propose to implement an FSM acceleration tool over the DG solver, avoiding the drawbacks encountered in Zhang et al. (2011). To make this acceleration possible, I propose to use a degenerate local solver to overcome transient non-stable states within each element. The new algorithm is based on the state-of-the-art DG scheme of Cheng and Wang (2014). Thus, the ℓ^2 reconstruction of spatial derivatives is avoided. Without any need of initial solution guess, features such as point source singularity treatment, 2D-TTI and 3D-TOR anisotropy, and complex topographies are integrated inside this new approach, yielding highly accurate traveltime estimation with a rather simple handling by potential users. In addition, the Runge–Kutta integration scheme is replaced by a local explicit Euler step. Thanks to this local, element by element, integration procedure, the constraint on the choice of the Hamiltonian discussed in section 2.2.2.1 is relaxed.

This new algorithm yields a drastic speed-up of the computation. A linear complexity is observed in the numerical experiments. Thanks to this improvement, going to three dimensions become accessible. The extension to three dimensions leads to an accurate and efficient solver for 3D traveltime computation, which I refer to as the FSM-DG solver.

Outline This chapter is organized as follows:

- In section 3.1, I present the numerical methods I develop for the integration of the fast sweeping inside the DG strategy.
- In section 3.2, I perform an efficiency analysis to highlight the significant gain provided by the FSM algorithm compared to the RK-DG strategy, and the competitiveness of the FSM-DG method when compared to more standard FSM-FD approaches. In particular, the linear complexity in $\mathcal{O}(N)$ of the FSM-DG algorithm is observed.
- In section 3.3, I exhibit the effectiveness of the FSM-DG strategy in complex media involving topographies.

The materials presented in this chapter are largely extracted from the article Le Bouteiller et al. (2018b), recently submitted.

3.1 Fast-sweeping algorithm and extension to 3D

3.1.1 Fast-sweeping algorithm

3.1.1.1 A global block-Gauss–Seidel strategy

In the RK-DG method presented in chapter 2, scheme (2.39) is solved over the whole domain in a time-marching approach. Instead, in the FSM approach, elements are updated sequentially, following specific orderings, to a temporary local stationary solution thanks to a local iterative strategy (local solver). Nonlinear equations are solved element by element, each element representing several unknowns (degrees of freedom), resulting in a block-Gauss–Seidel approach[1].

Formally I introduce the variable ξ_i as a local pseudo-time variable, which is no longer global. The resulting factored Eikonal equation (2.10) to solve inside each element for the isotropic case becomes

$$\partial_{\xi_i}\tau_i + \|\nabla_{\boldsymbol{x}} u_0 + \nabla_{\boldsymbol{x}}\tau_i\| - \frac{1}{c(\boldsymbol{x})} = 0, \tag{3.1}$$

where $\tau_i(\boldsymbol{x},\xi_i) = \tau_{|K_i}(\boldsymbol{x},\xi_i)$ is the solution inside element K_i. Similarly, in the 2D TTI case, the factored Eikonal equation (2.13) becomes

$$\begin{aligned}\partial_{\xi_i}\tau_i \;+\; & d[(u_{0,x}+\tau_{i,x})\cos\theta + (u_{0,z}+\tau_{i,z})\sin\theta]^2 + e[(u_{0,z}+\tau_{i,z})\cos\theta - (u_{0,x}+\tau_{i,x})\sin\theta]^2 \\ & + c[(u_{0,x}+\tau_{i,x})\cos\theta + (u_{0,z}+\tau_{i,z})\sin\theta]^2[(u_{0,z}+\tau_{i,z})\cos\theta - (u_{0,x}+\tau_{i,x})\sin\theta]^2 - 1 = 0.\end{aligned} \tag{3.2}$$

In the FSM approach, equations (3.1) and (3.2) are solved to the steady-state for each element, sequentially. They are solved using the DG scheme (2.39) which I recall here for a given element K_i, replacing ξ by ξ_i and u_h by u_i which denotes the numerical solution inside element K_i:

[1]Note that when implementing such an approach, it is no more possible to parallelize the local updates as in the RK-DG case, since these updates must be performed sequentially. However, this is not an issue in practice for geophysical applications which generally imply a lot of sources. The parallelization can be easily performed at the source level, by launching computations for different sources on different processors in an embarrassingly parallel way.

Find $u_i(.,\xi_i) \in \mathcal{P}_i,\ \forall \xi_i \geqslant 0$ such that

$$\begin{aligned}
&\int_{K_i} \Big(\partial_{\xi_i} u_i(\boldsymbol{x},\xi_i) + \mathcal{H}\big(\boldsymbol{x}, \nabla_{\boldsymbol{x}} u_i(\boldsymbol{x},\xi_i)\big)\Big) v_i(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\
&+ \int_{\partial K_i} \min\big(\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi_i), 0\big) [u_i](\boldsymbol{x},\xi_i) v_i^-(\boldsymbol{x}) \mathrm{ds} \\
- C\Delta K_i \sum_{S_i^j \in \partial K_i} \frac{1}{\Delta S_i^j} &\int_{S_i^j} \big(\chi_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi_i) - |\tilde{\mathcal{H}}_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi_i)|\big) [\nabla_{\boldsymbol{x}} u_i \cdot \boldsymbol{n}_{K_i}](\boldsymbol{x},\xi_i) v_i^-(\boldsymbol{x}) \mathrm{ds} \\
- 2C\Delta K_i \sum_{\bar{S}_i^j \in \bar{\partial} K_i} \frac{1}{\Delta \bar{S}_i^j} &\int_{\bar{S}_i^j} \min\big(\mathcal{H}^-_{\boldsymbol{n}_{K_i}}(\boldsymbol{x},\xi_i), 0\big)\ \big(\nabla_{\boldsymbol{x}} u_i^-(\boldsymbol{x},\xi_i) \cdot \boldsymbol{n}_{K_i}\big)\ v_i^-(\boldsymbol{x})\ \mathrm{ds} = 0,
\end{aligned} \tag{3.3}$$

for any test function $v_i \in \mathcal{P}_i$.

As in Zhao (2005), the four natural orderings of elements of the structured Cartesian mesh define the four alternating sweepings. For unstructured grids, it could be possible to pre-compute specific orderings of the elements for sweeping: I have not implemented this strategy because of the additional implementation complexity (Qian et al., 2007a,b). For Cartesian grids, the natural directions sample efficiently the characteristics of the Eikonal equation.

3.1.1.2 A local solver based on an explicit Euler method

Here I detail the local procedure for updating the solution inside a given element. Given current solutions in its neighbors, the solution inside an element evolves until its local steady state is reached, before considering the next element. Therefore, the local solver consists in solving scheme (3.3) repeatedly for a given element K_i. The local integration in ξ_i, which can be considered as a local fixed-point procedure, is performed with an explicit Euler method. Using an explicit time integration is advantageous because scheme (3.3) is highly nonlinear: the numerical fluxes depend on the solution itself in a nonlinear way. Moreover, I verify in my experiments that it is not necessary to use higher-order time schemes like second-order Runge–Kutta method in this local solver. In practice, the local solver implementation is straightforwardly derived from the RK-DG formulation, taking care of restricting the computation to the current element. Iterations are performed until the following inequality is satisfied:

$$\left\| \frac{u_i - u_i^{old}}{u_i^{old}} \right\| < \epsilon, \tag{3.4}$$

where u_i is the current solution inside element K_i while u_i^{old} is the solution at the previous Euler step, and ϵ is a convergence criterion. In practice, I consider the ℓ^2 norm and values of ϵ between 10^{-7} and 10^{-9}.

However, in some transient configurations, the convergence may not be reached due to the nonlinearity of the problem and causality considerations. When the upwind flux is initially oriented toward a given direction at an edge of a cell (e.g. an initial orientation given by a direct wave), the cell could be in a configuration such that changing its orientation to another direction (e.g. a diving wave coming through a higher velocity zone) turns out to be impossible during the fixed-point local procedure. Once the local solver has reached a given number of local iterations n_{max} without local convergence, I have designed a specific procedure, that I call *degenerate local solver*, which is activated for overcoming this

situation. This degenerate solver overwrites the solution $u_i(\boldsymbol{x})$ obtained after n_{max} iterations inside the current element K_i by a degenerate solution computed as follows. First, the degenerate solver detects, among the four neighbors of the current element K_i, the one with the lowest traveltime values. In practice I compare the maximum values at Gauss points on edges. The shortest traveltime value t_{min} is defined by

$$t_{min} = \min_{k \in \mathcal{N}_i} \left(\max_{\boldsymbol{x} \in s_{ik}} \left(u_k(\boldsymbol{x}) \right) \right), \tag{3.5}$$

where $\mathcal{N}_i$ denotes the set of indices of the neighbors of element K_i, and s_{ik} denotes the set of coordinates of Gauss points at the interface between elements K_i and K_k. The quantity t_{min} is obtained for a given neighbor that I denote $k^\star$. Once t_{min} has been computed, I simulate a (non-physical) plane wave coming from the edge between elements K_i and $K_{k^\star}$ and traveling through the element at the local wave speed. The traveltime degenerate solution inside element K_i then writes

$$u_i(\boldsymbol{x}) = t_{min} + \frac{\text{dist}_{k^\star}(\boldsymbol{x})}{c_i^\star}, \tag{3.6}$$

where the function $\text{dist}_{k^\star}(\boldsymbol{x})$ is the distance between the point $\boldsymbol{x}$ and the interface between element K_i and element $K_{k^\star}$, and where

$$c_i^\star = \min_{\boldsymbol{x} \in K_i} c(\boldsymbol{x}). \tag{3.7}$$

This plane-wave solution is then projected onto the DG local space $\mathcal{P}_i$. Considering the minimum velocity $c_i^\star$ ensures that if the velocity is not homogeneous inside element K_i, I do not introduce an information which is faster than the physical one. In my experiments, I observe that this is the best way to ensure stability when using this degenerate solver.

The temporary solution reconstructed in that way exhibits a correct orientation of fluxes and respects the isotropic Eikonal equation for the velocity $c_i^\star$ inside the element[2]. This will not be the final solution. This temporary simple estimation enables various branches of the final solution to propagate along the current sweep and eventually keeps the fastest one at each location. Indeed, I proceed with sweeps until this degenerate solver is not activated anymore: this is one of the criteria required for global convergence.

The use of a degenerate local stencil is not uncommon: it appears in FD solvers. The one I have designed for the DG framework is a key procedure which unlocks several crucial issues for developing an efficient FSM-DG method. First, it avoids to be trapped in a wrong causality setting, as described above. Second, as a consequence, it exempts from the need of a good initial solution, which was a practical limitation in Zhang et al. (2011). Finally, it exempts from a severe constraint on the Hamiltonian: the RK-DG scheme presented in chapter 2 requires a Lipschitz continuous Hamiltonian in order to define a suitable CFL condition which would not depend on the solution. This limitation prompts me to use the complicated 2D TTI Hamiltonian (2.38) in order to ensure stability in all the elements at every timestep. In the new FSM-DG method, if an instability raises during the local integration, then the degenerate solver acts as a post-treatment limiter, by replacing the solution with an approximate one respecting the local causality. Therefore, I am able to use the standard anisotropic equation (3.2) in the FSM-DG method.

[2]Note that in the anisotropic case, the degenerate solver is kept as such, considering the vertical velocity V_P for the computation of the degenerate solution.

3.1.1.3 Initialization

Initialization steps are defined for both local solver and sweeping procedure. Regarding the sweeping, the first four sweeps are performed from the source element toward the boundaries, respectively in the four quarters of the domain defined by the horizontal and vertical axes aligned with the source point. I have found this to be the best initialization in order to optimize the number of sweeps needed to reach the convergence. At the local level, the first time the local solver is called for a given element, the degenerate solver is executed first, then the local iterative procedure occurs. The element is tagged as updated afterwards. When executing the local solver, only the at-least-once updated neighbors are considered. At edges where a neighbor has not been updated yet, a boundary condition is applied. This condition is the same as the fourth term in scheme (3.3), applied at the domain boundaries.

3.1.1.4 FSM-DG algorithms

The pseudo-code of the numerical algorithms implemented for the FSM-DG method is given in algorithms 3.1 to 3.4. Note that in all these algorithms, the elements are indexed with two indices to locate them in the Cartesian grid.

The fast-sweeping procedure and its initialization are detailed in algorithms 3.1 and 3.2, respectively. To avoid infinite loops, a maximum number of global iterations (sweeps) is implemented (*max_iter*). Between two sweeps, the magnitude of the update is computed and compared to a convergence criterion *global_conv*. In the initialization step, a flag *updated* is initialized to a *false* value for all the elements. Once the local solver has computed an update for an element, the flag for this element is switched to a *true* value. The four initialization sweeps start at the source element indexed by (ix_s, iz_s).

The structure of the local solver is presented in algorithm 3.3. The first time the local solver is called for a given element, it computes an initial solution inside this element by calling the degenerate solver (line 3). The local solver performs an Euler step (line 11) in a direction given by the computation of the integrals of the DG scheme with appropriate functions: HJ_integral for the integral over the element (line 5), flux_integral for the integrals at interfaces between elements (line 8), boundary_integral for integrals at the domain boundary or at an interface between the current element and a neighboring element which has never been updated yet (line 10). This is the case when the *updated* flag for such a neighbor has a *false* value. The magnitude of the Euler update of the current element is computed and compared to the convergence criterion ϵ (line 12). A maximum number of Euler steps is defined with n_{max}. When this number is reached, the local degenerate solver is called to overcome transient non-converging local states (line 14).

Finally, the degenerate solver is detailed in algorithm 3.4. When the minimum traveltime value t_{min} has been found among the neighboring elements thanks to expression (3.5), a plane wave is simulated inside the current element and a projection of this solution is performed onto the DG approximation space.

Algorithm 3.1 Sweeping

```
1: procedure SWEEP1
2:     for i = 1 → N_x do
3:         for j = 1 → N_z do
4:             call LOCAL_SOLVER(i, j)
5: procedure SWEEP2
6:     for i = N_x → 1 do
7:         for j = 1 → N_z do
8:             call LOCAL_SOLVER(i, j)
9: procedure SWEEP3
10:    for i = N_x → 1 do
11:        for j = N_z → 1 do
12:            call LOCAL_SOLVER(i, j)
13: procedure SWEEP4
14:    for i = 1 → N_x do
15:        for j = N_z → 1 do
16:            call LOCAL_SOLVER(i, j)
17: procedure SWEEPING
18:    call INIT_SWEEP
19:    for k = 1 → max_iter do
20:        select case mod(k − 1, 4) + 1
21:            case(1) call SWEEP1
22:            case(2) call SWEEP2
23:            case(3) call SWEEP3
24:            case(4) call SWEEP4
25:        if ||(u − u^old)/u^old|| < global_conv then return
```

Algorithm 3.2 Init Sweeping

procedure INIT_SWEEP
  $updated(:,:)$=False
  for $i = ix_s \rightarrow N_x$ **do**
    for $j = iz_s \rightarrow N_z$ **do**
      call LOCAL_SOLVER(i, j)
      $updated(i, j)$=True
  for $i = ix_s \rightarrow 1$ **do**
    for $j = iz_s \rightarrow N_z$ **do**
      call LOCAL_SOLVER(i, j)
      $updated(i, j)$=True
  for $i = ix_s \rightarrow 1$ **do**
    for $j = iz_s \rightarrow 1$ **do**
      call LOCAL_SOLVER(i, j)
      $updated(i, j)$=True
  for $i = 1 \rightarrow ix_s$ **do**
    for $j = iz_s \rightarrow 1$ **do**
      call LOCAL_SOLVER(i, j)
      $updated(i, j)$=True

Algorithm 3.3 Local Solver

procedure LOCAL_SOLVER(i, j)
  if not $updated(i, j)$ **then**
    call DEGENERATE_SOLVER(i, j)
  for $n = 1 \rightarrow n_{max}$ **do**
    $inside$ = HJ_integral(i, j)
    for $(k, l) \in$ neighbors(i, j) **do**
      if $updated(k, l)$ **then**
        $edges \leftarrow edges$ + flux_integral(i, j, k, l)
      else
        $edges \leftarrow edges$ + boundary_integral(i, j, k, l)
    $u_{ij} = u_{ij}^{old} + \Delta t(inside + edges)$
    if $\left\| \frac{u_{ij} - u_{ij}^{old}}{u_{ij}^{old}} \right\| < \epsilon$ **then return**
    $u_{ij}^{old} = u_{ij}$
  call DEGENERATE_SOLVER(i, j)

Algorithm 3.4 Degenerate Solver

procedure DEGENERATE_SOLVER(i, j)
  $t_{\text{min}} = 100000$
  for $(k, l) \in$ neighbors(i, j) **do** $t_{\text{min}} = \min(t_{\text{min}}, \max(u_{kl}^{\text{edge}}))$
  $u_{ij} = \text{DG\_projection}\left(t_{\text{min}} + \frac{\text{dist}_{k\star}(\boldsymbol{x})}{c_{ij}^{\star}}\right)$

3.1.1.5 Illustration of the fast-sweeping procedure

In this section, I give an illustration of the sweeping procedure. For this illustration, the computation is performed inside a 4×4 km square. The point source is located at the center. The mesh is Cartesian, with P^1 rectangular elements and $N_x = N_z = 200$. Inside this domain, a background velocity is set to 2000 m/s, and two Gaussian velocity perturbations are added. One is positive, the other one is negative. The resulting model is shown in figure 3.1, together with the isocontours of the traveltime solution at the global steady-state. To obtain this global convergence, 14 sweeps are performed.

In figure 3.2, intermediate results obtained after successive sweeps are plotted. The initial solution corresponds to a homogeneous model of velocity 4000 m/s. The first four sweeps are not performed in the whole domain, but only in successive quarters of the domain, starting at the source element. These quarters are defined by the vertical and horizontal axes passing by the source. The degenerate solver is activated before each element update as an initialization tool during these four sweeps. In the first four plots, one can see the evolution of the solution in the four quarters, successively. The following sweeps are performed in the whole domain, in the four directions of the grid alternatively, starting at the four corners of the domain alternatively. The degenerate solver is now activated only if the local convergence is not reached. In the final result, characteristics circumvent the negative (top-left) velocity perturbation on both sides of it. After sweeps #3 to #6, around this perturbation, a short traveltime information coming from the bottom quarter is missing in the top quarter. During sweep #7, the sweeping orientation allows this information to propagate from the bottom to the top. After sweep #8, only few elements have not yet reached their final state. These elements are located in the complex region around the negative perturbation.

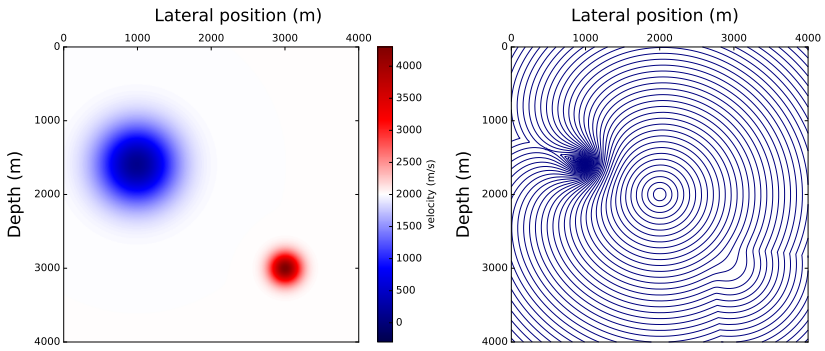


Figure 3.1: Illustration of the sweeping procedure. Velocity model (left) and isocontours of the final traveltime solution after 14 sweeps (right).

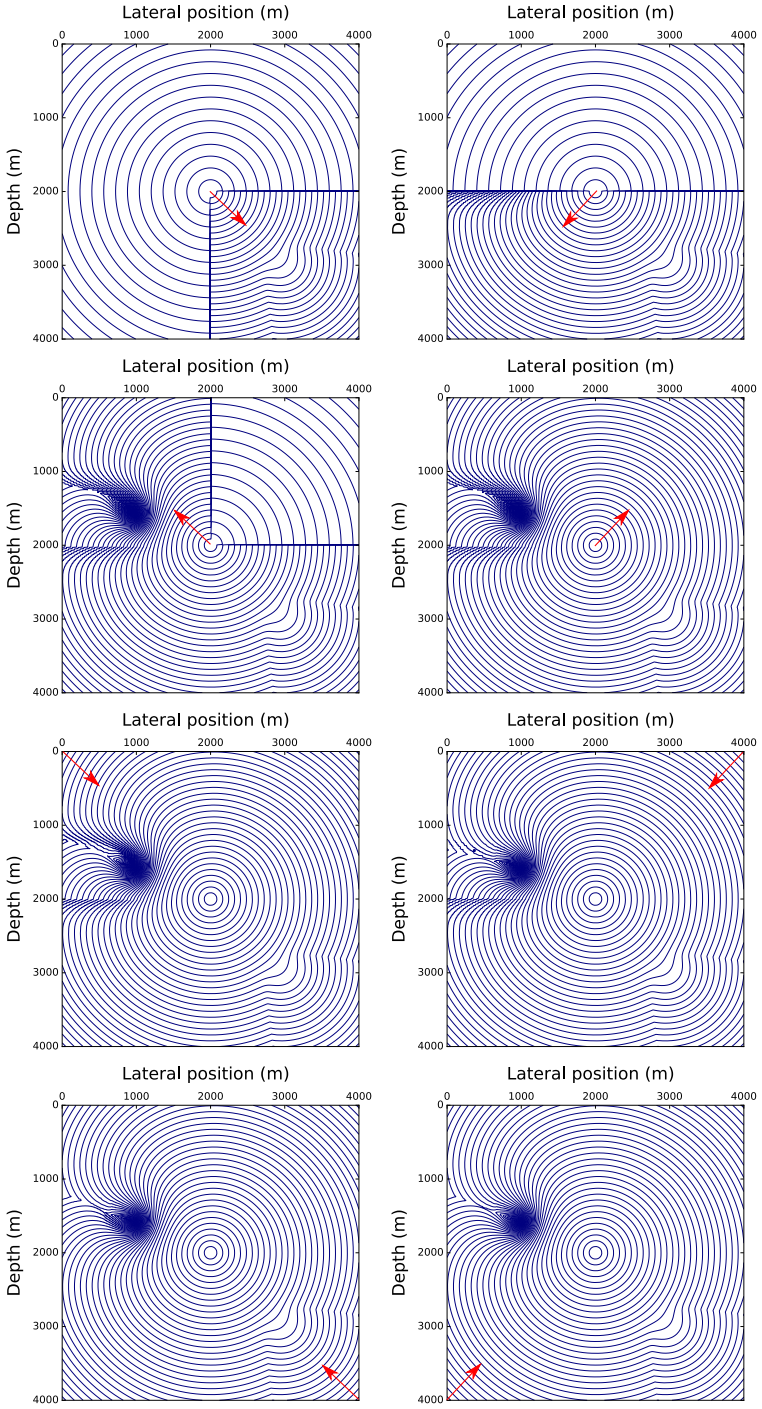


Figure 3.2: Illustration of the sweeping procedure. Isocontours of the traveltime solution after the first 8 sweeps, successively, from top to bottom, left to right. The red arrow indicates the starting point and the direction of the sweep.

3.1.2 Extension to 3D

The efficiency of the FSM-DG algorithm opens the door to 3D computation. Next, I present the modifications implied in terms of equations and implementation.

3.1.2.1 3D TOR Eikonal equation

The isotropic equation (3.1) can be directly used in a 3D configuration. For the 3D TOR configuration, traveltime verifies equation (1.48) with the rotation operator (1.53). For accuracy purpose, the point-source factorization is employed in the same way as before, which leads first to the following VOR Hamilton–Jacobi equation:

$$\begin{aligned}&\partial_{\xi_i}\tau_i + a(u_{0,x}+\tau_{i,x})^2 + b(u_{0,y}+\tau_{i,y})^2 + c(u_{0,z}+\tau_{i,z})^2\\ &+d(u_{0,x}+\tau_{i,x})^2(u_{0,y}+\tau_{i,y})^2 + e(u_{0,x}+\tau_{i,x})^2(u_{0,z}+\tau_{i,z})^2 + f(u_{0,y}+\tau_{i,y})^2(u_{0,z}+\tau_{i,z})^2\\ &+g(u_{0,x}+\tau_{i,x})^2(u_{0,y}+\tau_{i,y})^2(u_{0,z}+\tau_{i,z})^2 - 1 = 0.\end{aligned} \tag{3.8}$$

Then, the tilted TOR case is easily retrieved by substituting the spatial derivatives of τ_i and u_0 with the ones in the tilted referential. The reference solution u_0 might be chosen to be the analytical solution in homogeneous elliptical or anelliptical TOR medium, with elastic parameters given by their value at the source point.

3.1.2.2 3D implementation

The DG scheme (3.3) is written in a general formulation so that we may directly consider it for the 3D case. In practice, the main differences to implement in the FSM-DG algorithm are the followings:

- Normal and tangential vectors involved in scheme (3.3) are defined in 3 dimensions; one normal vector and two tangential vectors are defined per element face.
- In scheme (3.3), the integrals are calculated over cell volumes (3D) and cell surfaces (2D) instead of cell surfaces (2D) and cell edges (1D), respectively, with suitable Gauss points. An element shares faces with six neighbors. There is no difficulty to extend the 2D settings to the 3D settings in the Cartesian case. However, when a topography is introduced, a mesh deformation strategy is introduced. I describe it in section 3.1.2.3.
- Basis functions are extended to 3D, keeping the same principles from Piperno (2005). The basis functions I use for 3D P^1 approximation are given in appendix 3.A.
- The sweeping strategy consists of repeating 8 sweeps instead of four, alternatively, in the eight natural directions of the 3D Cartesian mesh. Similarly, eight initialization sweeps are applied successively to the eight subdomains defined by the three main axis aligned with the source point.
- The six neighbors instead of four have to be considered in the degenerate solver.

The algorithms are easy to extend to the 3D case.

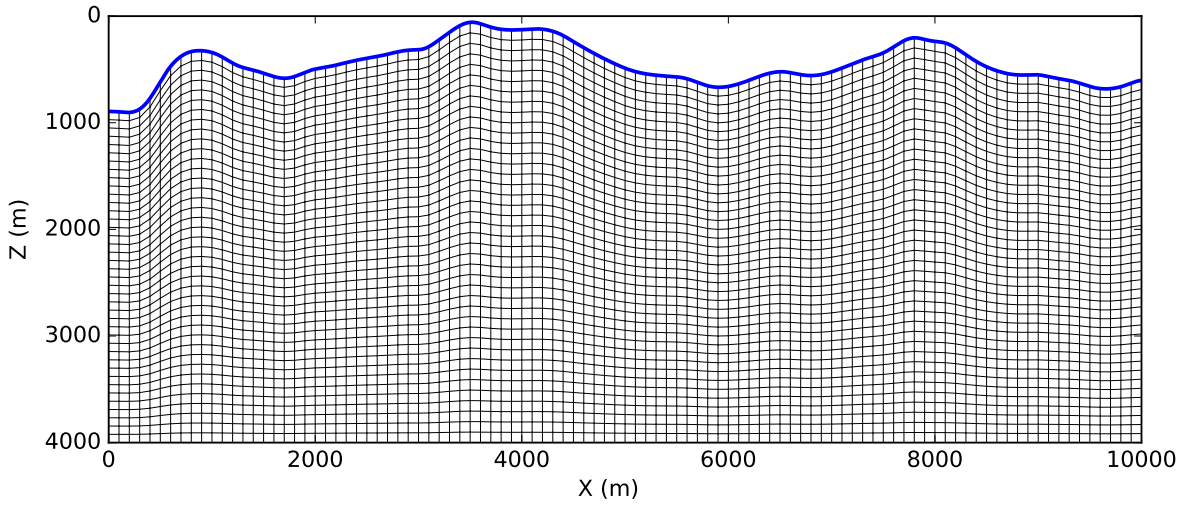


Figure 3.3: Vertically deformed mesh for a cross-section extracted from SEAM II Foothills benchmark model. The topography (blue line) has been interpolated over 101 equally-distributed points in x-direction. The mesh is 100×40 elements.

3.1.2.3 Mesh deformation for topography

In presence of a non-flat topography, building an unstructured mesh as in section 2.4.2 could be performed in 3D. However, as mentioned in section 3.1.1.1, I have not implemented this kind of strategy for the sake of efficiency: it would imply the loss of natural orderings in the mesh.

Instead, at the exploration scale, vertically deformed Cartesian grids are very attractive for their simplicity (Tarrass et al., 2011; Trinh et al., 2018). Figure 3.3 shows an example of a vertically deformed mesh in 2D. The main advantages of this approach are that there is no mesh generation step to be performed by the user: the spatial position of each element can be accessed directly by its indices in x, y, and z. Moreover, the natural ordering of elements is kept when performing the FSM algorithm, yielding an optimal efficiency for the Gauss–Seidel procedure. The topography variation is simply described by a gradual vertical deformation of elements, keeping a constant number of elements in x, y, and z directions. After interpolating the topography $z^\star$ over the (x, y) nodes, the explicit mapping between a reference unit cube and each deformed element is established. This mapping is detailed in appendix 3.B. The basis functions, as well as the Gauss points used for computing integrals in scheme (3.3), are defined in the reference cube. The quantities in the physical space are estimated at these Gauss points using the mapping, the local Jacobian, and chain rules for the spatial derivatives.

In practice, the needed quantities at Gauss points, such as local Jacobian, values of basis functions and their derivatives, etc. are precomputed prior to solving the Eikonal equation itself, and only once for a given deformed grid.

3.2 Efficiency analysis

In this section, I illustrate the efficiency of the FSM-DG scheme in a smooth isotropic medium where the exact solution is known. I compare the CPU times required by the RK-DG, the FSM-DG and an FSM-FD method, as well as the corresponding ℓ^2 errors, in 2D and 3D cases.

RK-DG						
N	#dof	ℓ^2 error	Order	dx ℓ^2 error	Order	CPU (s)
21	1323	$1.74\text{E}-03$		$1.13\text{E}-02$		0.36
41	5043	$4.66\text{E}-04$	1.97	$5.51\text{E}-03$	1.08	1.6
81	19683	$1.21\text{E}-04$	1.98	$2.71\text{E}-03$	1.04	11
161	77763	$3.07\text{E}-05$	1.99	$1.34\text{E}-03$	1.02	78
321	309123	$7.74\text{E}-06$	2.00	$6.69\text{E}-04$	1.01	600
641	1232643	$1.94\text{E}-06$	2.00	$3.34\text{E}-04$	1.00	4900

Table 3.1: RK-DG results from the first numerical example in the 2D case. Number of elements along one direction (N), number of degrees of freedom (#dof), ℓ^2 error of the solution and its derivative along x-direction, convergence orders, and CPU times, for P^1 polynomial approximation.

FSM-DG							
N	#dof	ℓ^2 error	Order	dx ℓ^2 error	Order	CPU (s)	CPU ratio
21	1323	$1.73\text{E}-03$		$1.18\text{E}-02$		0.21	1.7
41	5043	$4.65\text{E}-04$	1.96	$5.60\text{E}-03$	1.11	0.43	3.7
81	19683	$1.21\text{E}-04$	1.98	$2.73\text{E}-03$	1.06	0.68	16
161	77763	$3.07\text{E}-05$	1.99	$1.35\text{E}-03$	1.03	2.0	39
321	309123	$7.74\text{E}-06$	2.00	$6.70\text{E}-04$	1.01	6.6	91
641	1232643	$1.95\text{E}-06$	1.99	$3.35\text{E}-04$	1.00	26	188

Table 3.2: FSM-DG results from the first numerical example in the 2D case. Number of elements along one direction (N), number of degrees of freedom (#dof), ℓ^2 error of the solution and its derivative along x-direction, convergence orders, CPU times, and CPU ratio between RK-DG and FSM-DG, for P^1 polynomial approximation. Please note the slow increase of the CPU time with respect to the number of degrees of freedom (see figure 3.4.b).

3.2.1 2D study

I first exhibit the efficiency of the FSM-DG technique in a 2D isotropic case, where the velocity varies linearly with depth. In a 4×4 km square, the velocity varies from 2 km/s at the surface to 4 km/s at depth. The point source is located at the surface with coordinates $x_s = 2$ km, $z_s = 0$ km. An error computation for traveltime as well as its spatial derivatives is performed thanks to the knowledge of the exact solution (see section 2.3.1). A refinement study is carried out and the results are shown in tables 3.1 and 3.2: ℓ^2 errors are exhibited with respect to the number of degrees of freedom when using P^1 polynomial approximations, together with a comparison of CPU times between the RK-DG and the FSM-DG techniques. I perform similar experiments using the isotropic Eikonal fast-sweeping FD solver from Noble et al. (2014); results are given in table (3.3). All these results are compared in figure 3.4, and I detail them next.

The convergence of the schemes are highlighted in figure 3.4.a. The conclusions in terms of convergence are the same as in section 2.3.1: The FD method exhibits a first-order convergence, which means that when the spatial discretization step is divided by two, the error is also divided by two. Note that both RK-DG method and FSM-DG method yield the same error, since they yield the same final state in a given discretization after integration in ξ. As expected, the higher slope of decrease of the error with respect to the number of degrees of freedom highlights the second-order convergence of the P^1 DG approximation. Consequently, the x-derivative of the solution exhibits a first-order convergence,

FSM-FD				
N	#dof	ℓ^2 error	Order	CPU (s)
72	5184	6.08E − 03		0.01
140	19600	2.91E − 03	1.06	0.05
278	77284	1.46E − 03	0.99	0.18
556	309136	7.32E − 04	1.00	0.76
1110	1232100	3.67E − 04	1.00	3.1
2220	4928400	1.84E − 04	1.00	12
4440	19713600	9.42E − 05	0.97	50

Table 3.3: FSM-FD results from the first numerical example in the 2D case. Number of elements along one direction (N), number of degrees of freedom (#dof), ℓ^2 error of the solution, convergence orders, and CPU times.

RK-DG						
N	#dof	Error	Order	dx error	Order	CPU (s)
11	5324	1.41E − 03		2.06E − 02		2.4
21	37044	3.73E − 04	2.05	1.11E − 02	0.96	4.3
41	275684	9.59E − 05	2.03	5.76E − 03	0.98	15
81	2125764	2.43E − 05	2.01	2.93E − 03	0.99	133
161	16693124	6.07E − 06	2.02	1.48E − 03	0.99	1150

Table 3.4: FSM-DG results from the 3D numerical example. Number of elements along one direction (N), number of degrees of freedom (#dof), ℓ^2 error of the solution and its derivative along x-direction, convergence orders, and CPU times, for P^1 polynomial approximation.

as shown in tables 3.1 and 3.2. In terms of computational cost, the analysis of the slopes in figure 3.4.b shows that the RK-DG method has a computational complexity in $\mathcal{O}(\#\text{dof}^{\,3/2})$, compared to the linear complexity in $\mathcal{O}(\#\text{dof})$ of the FSM-FD method. This is why the RK-DG method is not efficient. The new FSM-DG algorithm constitutes a huge improvement since it exhibits a linear complexity in $\mathcal{O}(\#\text{dof})$, as the FSM-FD scheme. Finally, the analysis of figure 3.4.c underlines that, for reaching low levels of error, the most efficient algorithm is the new FSM-DG. For a given level of error, this algorithm is much faster than FSM-FD because it needs less degrees of freedom to reach the same accuracy.

3.2.2 3D study

Keeping the vertical gradient of velocity, I now perform similar simulations on a 3D $4 \times 4 \times 4$ km cube with the FSM-DG method. The point source is located at the surface with coordinates $x_s = 2$ km, $y_s = 2$ km, $z_s = 0$ km. The results obtained with a P^1 approximation are detailed in table 3.4 and highlighted in figure 3.5. As expected, I retrieve a second-order convergence of the 3D DG discretization (figure 3.5.a). Moreover, the FSM-DG method still exhibits a linear complexity (figure 3.5.b), which makes it very efficient in 3D as well.

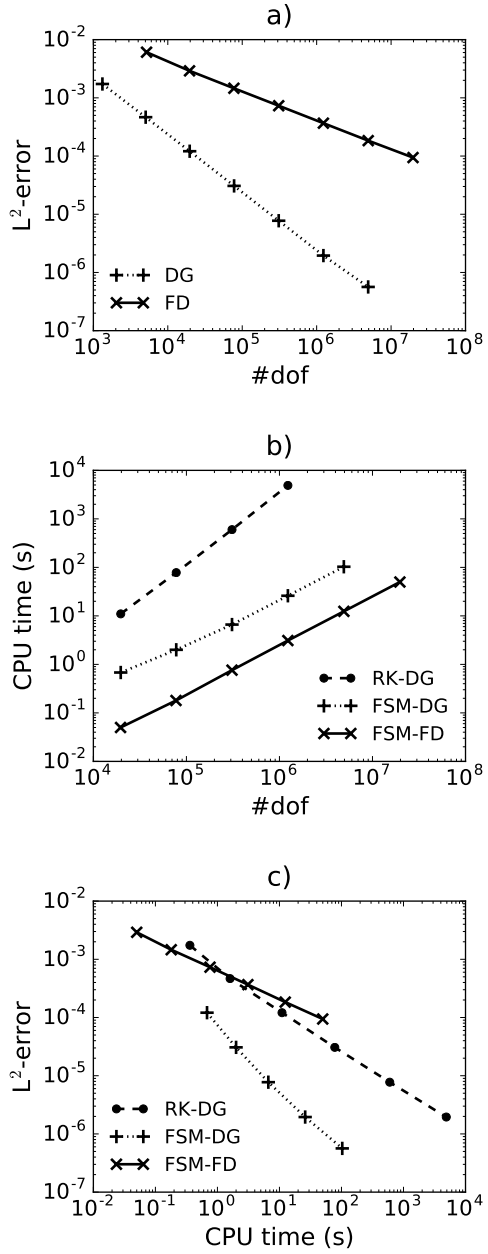


Figure 3.4: Comparison of FD and DG methods with respect to accuracy (ℓ^2 error), CPU time and number of degrees of freedom. (a) The DG scheme yields a high-order approximation, resulting in a lower error for a fixed number of degrees of freedom compared to the FD method. (b) Although the DG method implies a higher computational burden for a fixed number of degrees of freedom compared to the FD method, the fast-sweeping algorithm applied to the DG method (FSM-DG) exhibits a linear complexity, which is more efficient than the RK-DG method. (c) Finally, the resulting efficiency is higher with the FSM-DG method than the FD method: reaching a fixed level of error is cheaper.

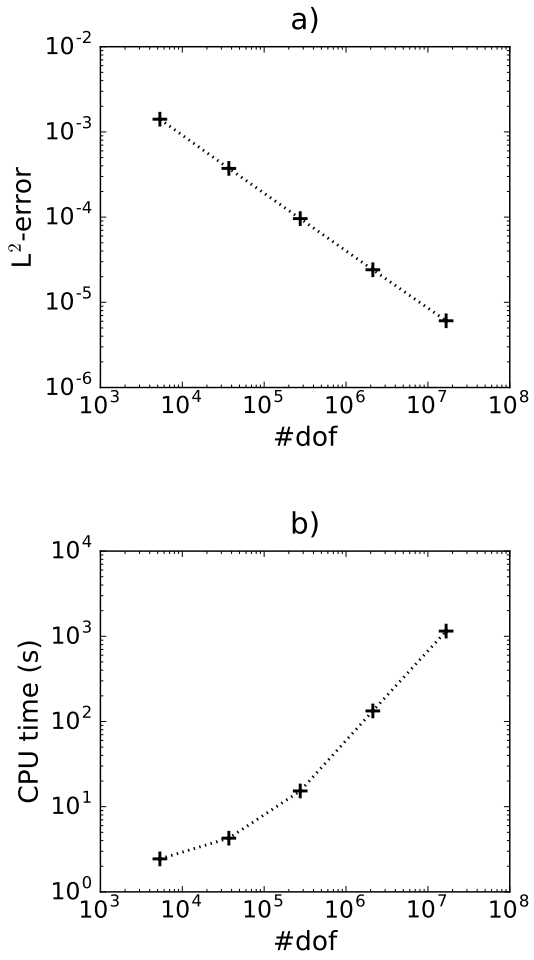


Figure 3.5: FSM-DG results from the 3D numerical example: ℓ^2 error and CPU time with respect to the number of degrees of freedom. The curved shape for small number of degrees of freedom when the CPU time is lower than 10 seconds is explained by initialization steps in the code that do not depend on the size of the mesh, such as model reading or source handling for instance.

3.3 Numerical illustrations

In this section I illustrate the potential of the FSM-DG method on two complex 3D examples:

1. A tilted orthorhombic media with a complex topography;
2. A complex realistic isotropic velocity model with a complex topography.

3.3.1 Homogeneous 3D tilted orthorhombic medium with complex topography

In this example, I consider a homogeneous 3D TOR medium, with anisotropic parameters chosen as follows:

$$\begin{cases} V_P = 2000\,\text{m/s}, \\ \epsilon_2 = 0.2, \delta_2 = 0.1, \\ \epsilon_1 = 0.4, \delta_1 = 0.3, \\ \delta = 0.1, \\ \phi = 30^\circ, \\ \theta = 45^\circ, \\ \psi = -15^\circ. \end{cases} \tag{3.9}$$

I define a physical domain with a complex topography using a part of the SEAM II model (Regone et al., 2017). I consider a domain of size 3 km along the x-axis, 7 km along the y-axis, and 2 km along the z-axis. The domain exhibits a spatially varying topography. I deliberately amplify by a factor 2 this vertical deformation of the surface in order to clearly exhibit its imprint on the traveltime. I build a vertically-deformed mesh from a regular Cartesian grid to follow this topography. The mesh spacing before deformation is 100 m in the three directions. The traveltimes are computed both with and without the topography, for a point-source located at $x = 1150$ m, $y = 50$ m, $z = 945$ m. Both results are superimposed in figure 3.6. The imprint of the topography is neat in the near-surface areas and might be understood by applying the Huygens principle at the bottom points of the topography, from which the upper parts of the domain are illuminated. This example shows the importance of taking care of topography, and the good behavior of the FSM-DG method when doing so. Let me emphasize that not only the traveltime is modified by the topography, but also its spatial derivatives, which indicate the front propagation direction. These derivatives are of major interest when considering subsequent amplitude or angle estimation.

3.3.2 3D complex SEAM II model

In this example, I apply the FSM-DG scheme onto the same part of the isotropic SEAM II model as before. Here, I consider the original topography as provided with the model (no amplification factor). To do so, I build a vertically-deformed mesh as in the previous example: The original mesh step is 100 m in the three directions, and the resulting deformed mesh is shown in figure 3.7. The P-wave velocity model, shown in figure 3.8, exhibits complex velocity structures with layering and faulting, as well as near-surface low-velocity areas which are known to impair imaging and inversion results if not carefully considered. I use the P-wave velocity of the model to compute the first-arrival traveltime for a source located at $x = 375$ m, $y = 350$ m, and $z = 528$ m. The traveltime isocontours are superimposed over the P-wave velocity model in figure 3.8. In order to illustrate the quality of the

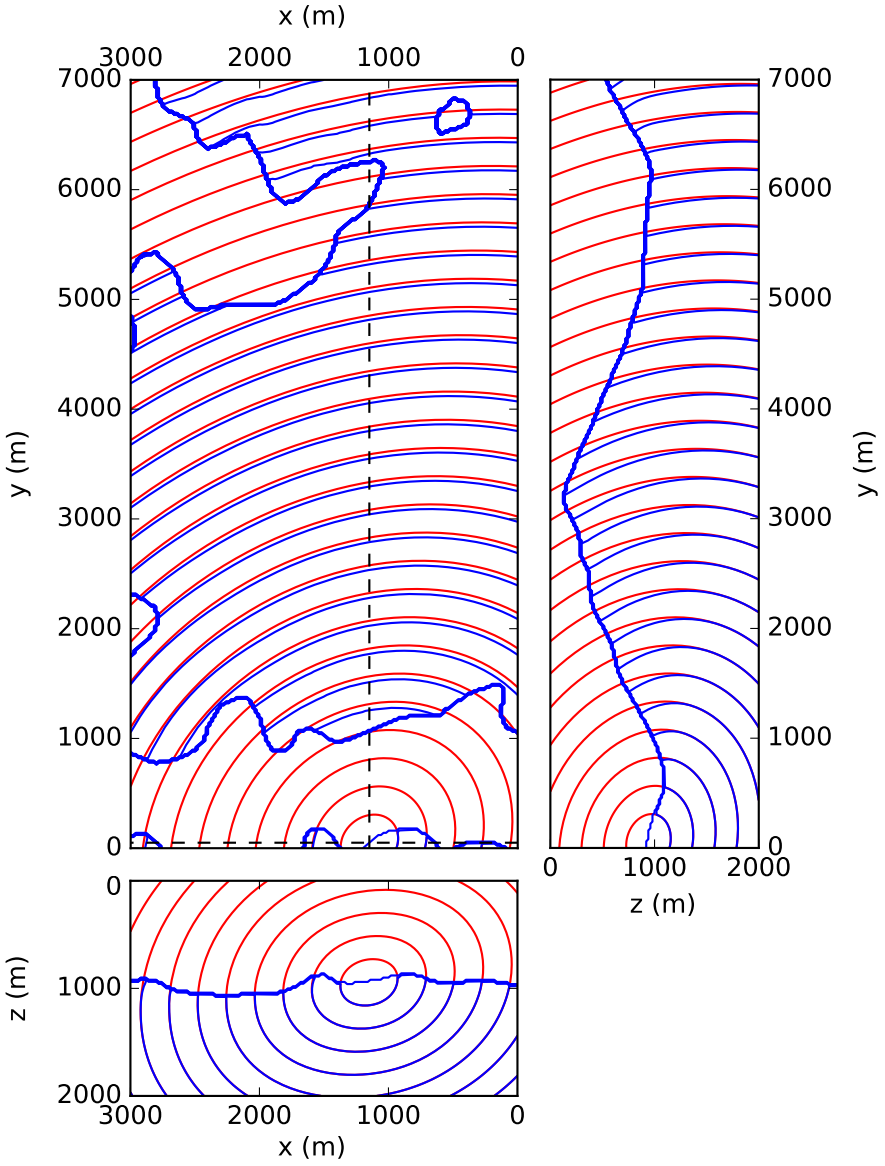


Figure 3.6: Traveltime isocontours computed in the homogeneous TOR model with model parameters given in expressions (3.9), with (blue) and without (red) the topography. Panels at $z = 945$ m (left), $x = 1150$ m (right), and $y = 50$ m (bottom). Please note that the thick zone is the contour of the topography: the blue solution is not built above the topography, while the red solution assumes a flat topography at the top of the domain. In the y-profile, the imprint of the topography is less visible than in the x-profile because of the flatter topography.

result, I perform a computation using a spectral-element solver for the elastic wave equation applied to the same model. For this computation I use the code SEM46 (Trinh et al., 2017), using an impulse source with a $60 \times 140 \times 40$ mesh. In figure 3.9 and 3.10, the traveltime isocontours obtained by the FSM-DG method are superimposed over the displacement field in the x-direction obtained with the SEM46 code, in vertical and horizontal planes, highlighting an almost perfect agreement between the first-arrival traveltime and the wavefront, although they are based on different equations (Eikonal equation versus elastic wave equation, respectively) and different numerical methods (discontinuous

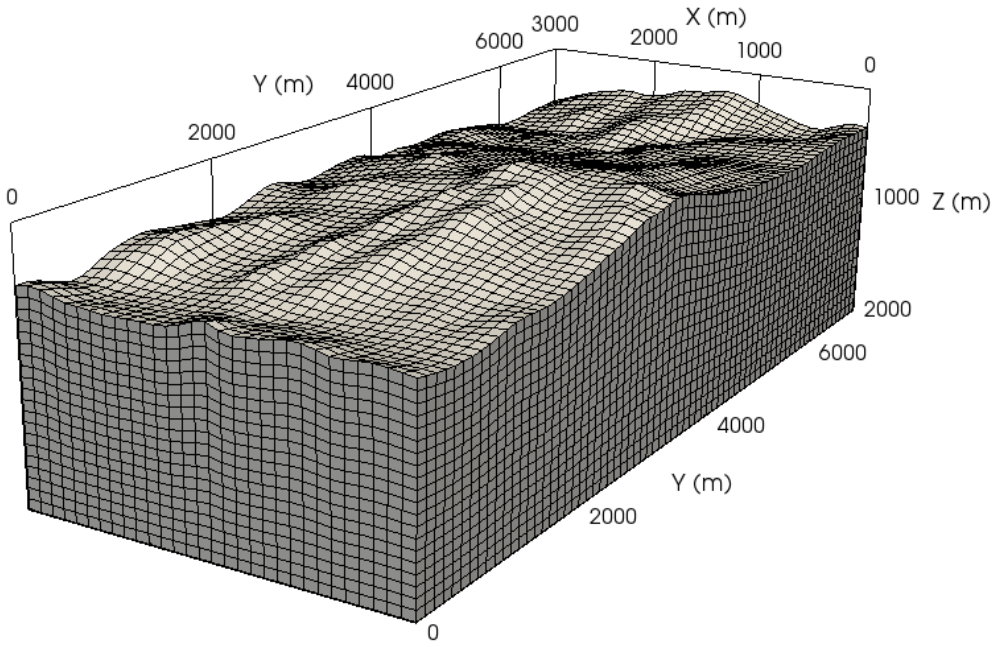


Figure 3.7: The 30x70x20 vertically-deformed mesh built for the SEAM II model.

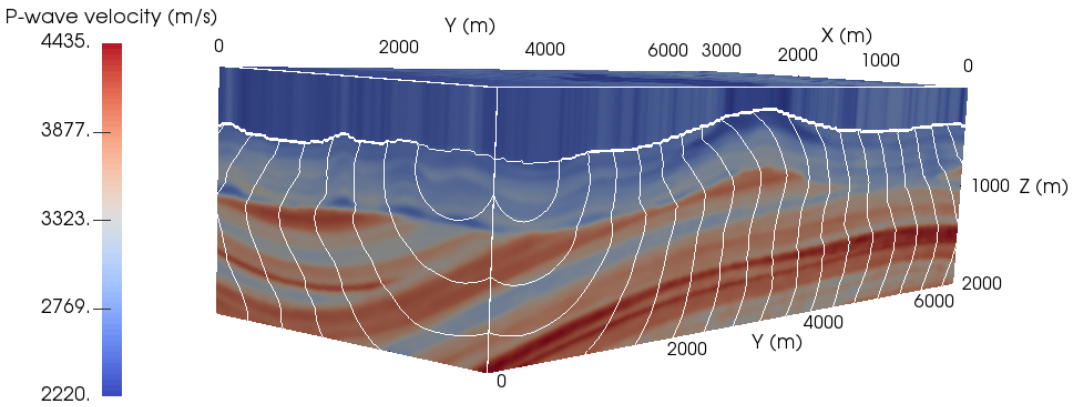


Figure 3.8: The Vp model extracted from the SEAM II model, with traveltime isocontours superimposed.

Galerkin method versus spectral element method, respectively).

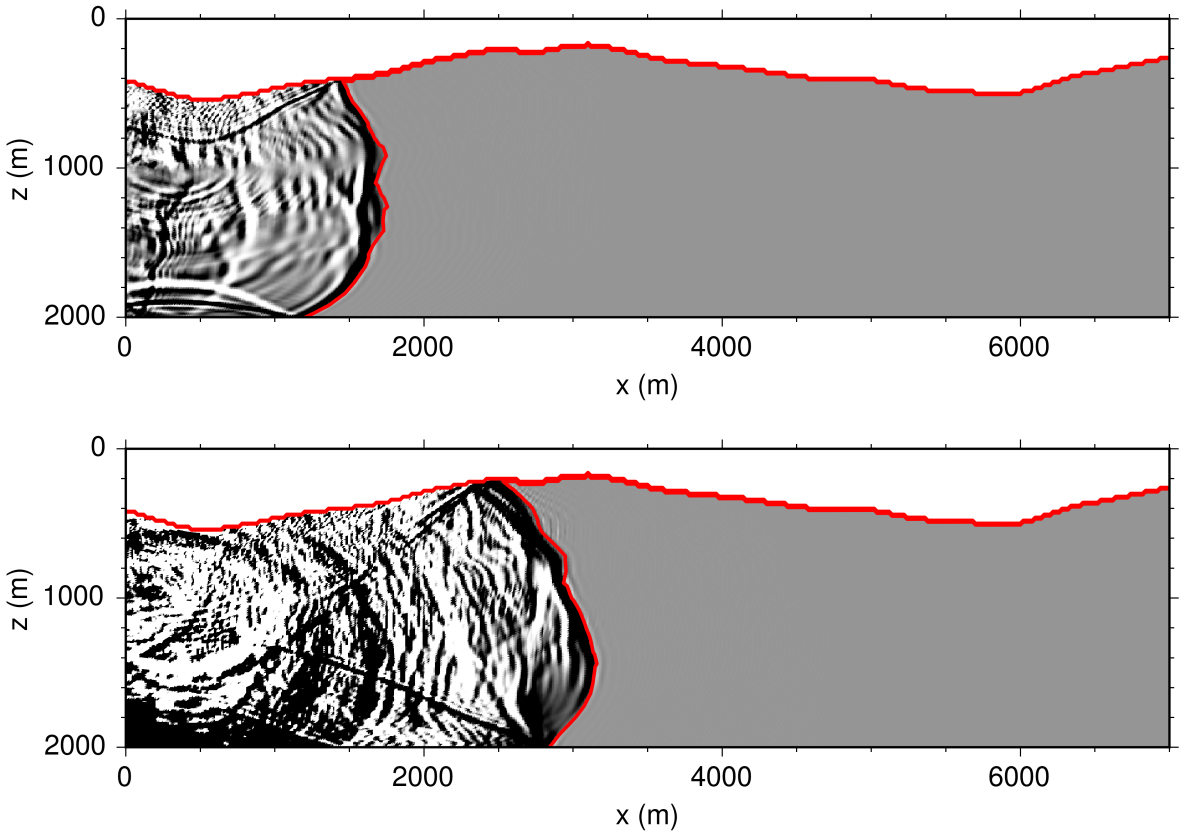


Figure 3.9: Displacement field in the x-direction from elastic spectral-element solver with an impulse source, with traveltime isocontours superimposed. Snapshots in the vertical plane $x = 1500$ m and corresponding isocontours at time 0.6 (top) and 0.9 (bottom) second. One could notice the numerical noise occurring ahead of the front, which comes from the wave propagation solution, and which is visible due to the saturation of the plot.

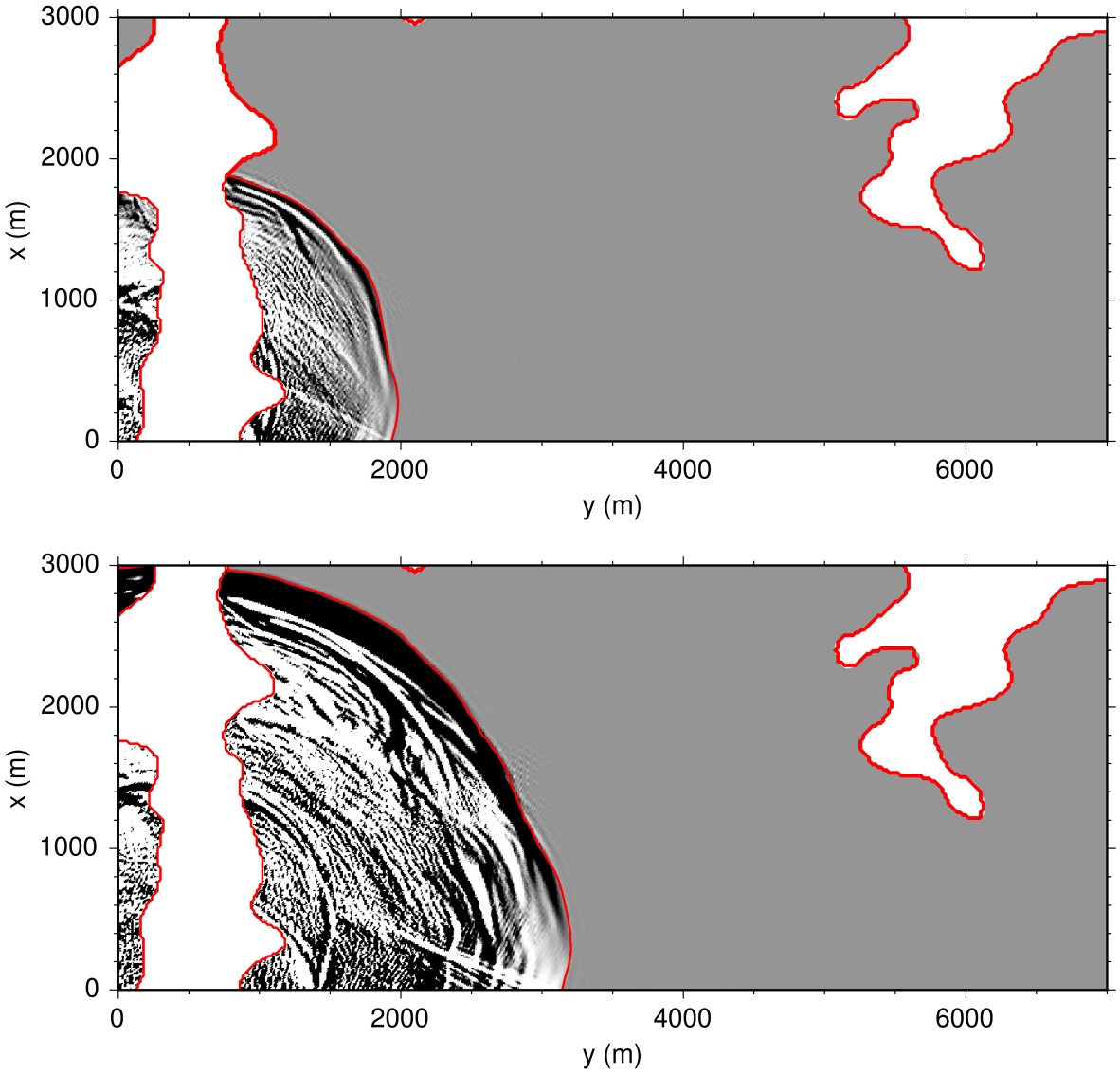


Figure 3.10: Displacement field in the x-direction from elastic spectral-element solver with an impulse source, with traveltime isocontours superimposed. Snapshots in the horizontal plane $z = 500$ m and corresponding isocontours at time 0.6 (top) and 0.9 (bottom) second. The topographical contour is expressed by the white zone where no solution is computed. Here again, one could notice the numerical noise occurring ahead of the front.

3.4 Conclusion

In this chapter, I have introduced an FSM-DG approach for solving the 3D Eikonal equation. The DG scheme had been previously introduced in chapter 2 in a 2D framework, using a Runge–Kutta solver, responsible for a high computational complexity in $\mathcal{O}(N^{3/2})$. The new FSM algorithm provides a significant increase of efficiency, making possible to reach a linear complexity in $\mathcal{O}(N)$ as for FSM-FD approach, while benefiting for the high accuracy and higher-order convergence rate associated with DG approach. For this reason, higher accuracy for the traveltime solution and its spatial derivatives is obtained compared to FD methods, while complex geometrical structures are handled in a stable and accurate way, thanks to FE properties. This is illustrated by the use of deformed Cartesian grid for handling topography. Both 2D and 3D implementations are performed, with TTI and TOR anisotropy, thanks to the general Hamiltonian formulation of the DG scheme. Even more general Hamiltonian could be considered in the future, accounting for instance for triclinic anisotropy.

The essential ingredient in the design of the FSM-DG solver is the degenerate local solver, which overcomes transient non-stable states at the local scale, while keeping the causality verified. This is a pragmatic, limiter-like way to eventually reach a global convergence, which I built based on my experience. Although I have no formal proof of convergence of this tool, it performs well in all the cases I have tried.

This degenerate local solver is also a key ingredient used before the first update of each element to initialize the solution, making it unnecessary to provide an accurate initial guess of the solution.

Now that I have designed an efficient and accurate tool for traveltime computation in complex media, I will present how to derive accurate numerical solvers for take-off angle, amplitude, and adjoint-state variable, which are useful in order to be applied to asymptotic inversion (traveltime/slope tomography) and imaging approaches (quantitative migration using amplitude and angle).

Appendices

3.A Basis functions in 3D geometry

In the 3D rectangular case, four fundamental basis functions are defined as

$$\begin{aligned}\phi^1(\xi,\eta,\zeta) &= \xi,\\ \phi^2(\xi,\eta,\zeta) &= \eta,\\ \phi^3(\xi,\eta,\zeta) &= \zeta,\\ \phi^3(\xi,\eta,\zeta) &= 1-\xi-\eta-\zeta.\end{aligned} \tag{3.10}$$

A P^1 element employs these three basis functions to span the space of first-order polynomials and to describe the numerical solution in this space. Higher-order basis functions can be derived from Piperno (2005). However, I do not detail them here, because I have only implemented and used P^1 approximation in 3D.

Like in 2D, a natural formulation for the rectangular case would involve more basis functions. This number would be 8 in 3D. Again, I choose to keep only the four basis functions in (3.10), and this choice is motivated by numerical experiments: I have tested both implementations and I verified that the convergence orders computed in the convergence analysis in section 2.3 are the same in both cases. Moreover, taking four fundamental functions instead of eight yields a quite significant decrease of the computational cost.

3.B Integral computation in 3D deformed elements

In this appendix, I give the mapping allowing to compute integrals in a reference unit cube for rectangular vertically deformed elements. A generic element K_i of the mesh is shown in figure 3.11, and its vertices are labeled from #1 to #8. I denote by $(x_i^j, y_i^j, z_i^j)_{1\leqslant j\leqslant 8}$ the physical coordinates of these vertices. The mapping from the reference coordinates to the physical coordinates is the following:

$$\begin{array}{rccl}\gamma_i & : & \hat{K} & \longrightarrow & K_i\\ & & (\xi,\eta,\zeta) & \longmapsto & (x_i,y_i,z_i),\end{array} \tag{3.11}$$

with

$$\begin{aligned}x_i =& x_i^1 + (x_i^2 - x_i^1)\xi,\\ y_i =& y_i^1 + (y_i^3 - y_i^1)\eta,\\ z_i =& \left(z_i^4(1-\zeta)+z_i^8\zeta\right)\xi\eta + \left(z_i^2(1-\zeta)+z_i^6\zeta\right)\xi(1-\eta)\\ &+ \left(z_i^3(1-\zeta)+z_i^7\zeta\right)(1-\xi)\eta + \left(z_i^1(1-\zeta)+z_i^5\zeta\right)(1-\xi)(1-\eta).\end{aligned} \tag{3.12}$$

The straightforward computation of the partial derivatives of the physical coordinates with respect to the reference coordinates yields the Jacobian

$$\begin{aligned}|J_i| =& \big((z_i^8 - z_i^4)\xi\eta + (z_i^6 - z_i^2)\xi(1-\eta)\\ &+ (z_i^7 - z_i^3)(1-\xi)\eta + (z_i^5 - z_i^1)(1-\xi)(1-\eta)\big)(x_i^2 - x_i^1)(y_i^3 - y_i^1).\end{aligned} \tag{3.13}$$

The inverse mapping writes

$$\begin{array}{rccl}\gamma_i & : & K_i & \longrightarrow & \hat{K}\\ & & (x_i,y_i,z_i) & \longmapsto & (\xi,\eta,\zeta),\end{array} \tag{3.14}$$

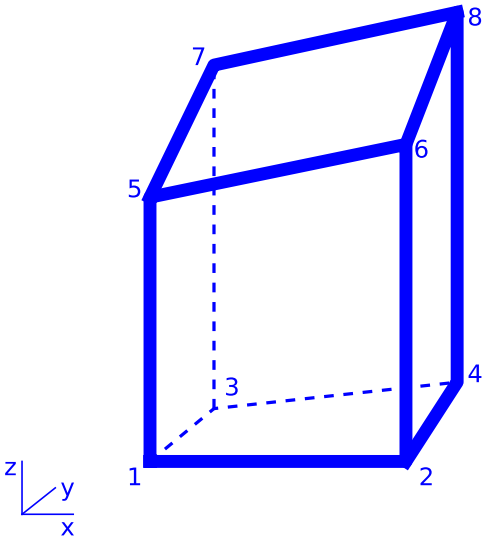


Figure 3.11: A vertically deformed element of the physical mesh. Note that edges between vertices #1 and #5, #2 and#6, #3 and #7, and #4 and #8 are vertical, since the mesh deformation occurs only vertically.

with

$$\begin{aligned}\xi =& \frac{x_i - x_i^1}{x_i^2 - x_i^1},\\ \eta =& \frac{y_i - y_i^1}{y_i^3 - y_i^1},\\ \zeta =& \frac{1}{|J_i|}\Big(z_i - \big(z_i^4\xi\eta + z_i^2\xi(1-\eta) + z_i^3(1-\xi)\eta + z_i^1(1-\xi)(1-\eta)\big)\Big).\end{aligned} \tag{3.15}$$

Computing integrals involving basis functions in the physical space and their derivatives is made possible by using basis functions in the reference space and making a change of variables. Like in the 2D case, this can be done easily by computing the inverse Jacobian matrix and using chain rules.

Chapter 4

Computation of take-off angle, amplitude, and adjoint-state variable

Contents

Introduction

I have shown in chapter 3 that it is possible to compute seismic traveltime by solving the Eikonal equation with an FSM-DG solver in an accurate and efficient way. Traveltime is useful as such, but it also can be combined with other asymptotic quantities in most of the conventional imaging and inversion strategies. In this chapter, I show how to use the FSM-DG solver and the associated traveltime in order to build efficient solvers for other quantities.

The quantities I consider are the followings:

- **take-off angle**: essential for localization of microseismic events, inversion of focal mechanisms, slope tomography, and for advanced imaging algorithms;
- **amplitude**: useful for reconstruction of asymptotic Green's functions and seismograms together with traveltime, also useful for imaging purpose;
- **adjoint-state variable**: in the case of traveltime tomography, the adjoint-state method requires to solve a transport equation, similar to the one for amplitude.

Interestingly, these three quantities verify PDEs involving the derivatives of the traveltime, as detailed in section 1.2. Traveltime computation is therefore a prerequisite for the computation of these derived quantities. Extracting these traveltime derivatives in a consistent numerical way is essential for the consistency of the discretization of these different PDEs.

The PDEs verified by these quantities are linear. Therefore, I could have chosen to use an explicit solver in the numerical algorithms. Instead, I choose to re-employ the efficient FSM procedure and the local Euler integration to keep a similar code implementation among all the solvers.

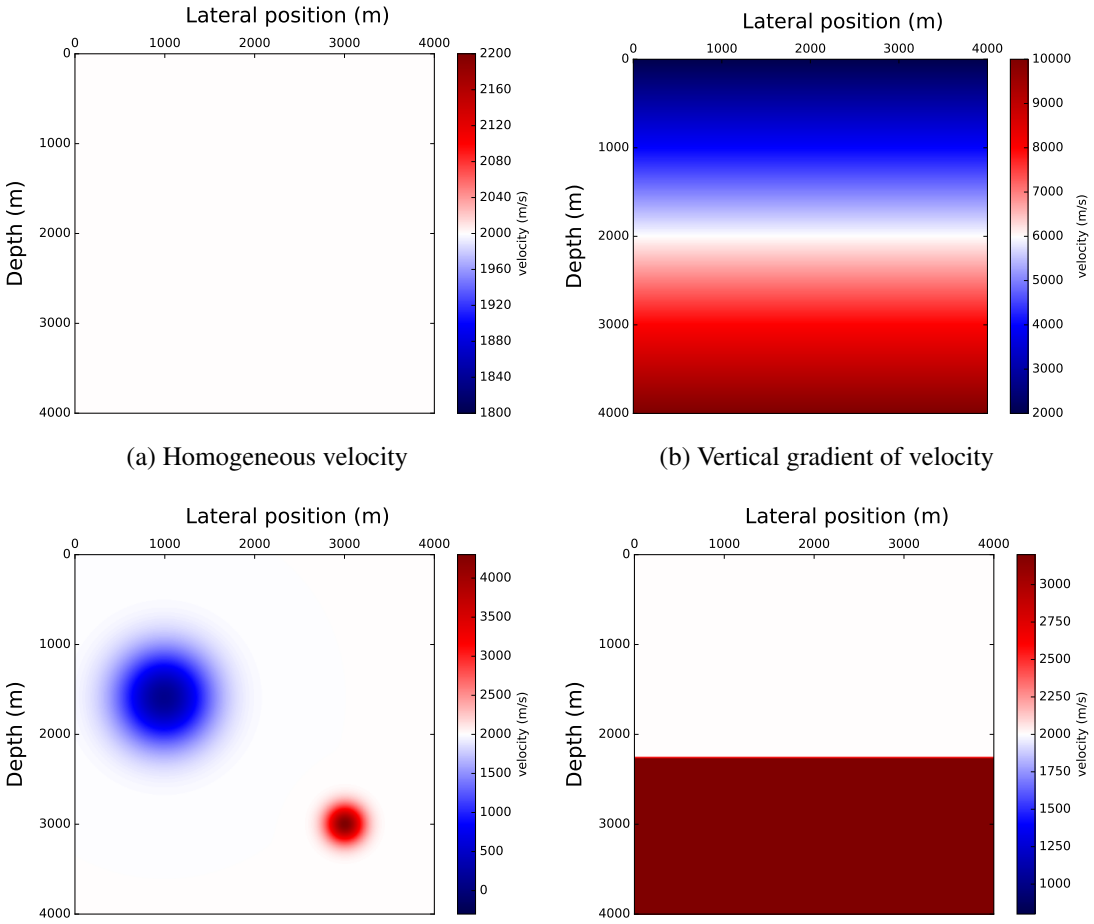
The outline of this chapter is the following:

- In section 4.1, I present an FSM-DG scheme, similar to the one used for traveltime, for the computation of take-off angle. I exhibit numerical illustrations on simple media.
- In section 4.2, I present a finite-volume (FV) approach derived from the FSM-DG method. FV methods are closely linked to DG methods and can be viewed as a lower-order formulation of DG using piecewise constant approximations. The FSM-FV method I derive is suitable for solving the transport equation. I show how to use it for the computation of amplitude and adjoint-state variable. I exhibit numerical illustrations on simple media.
- In section 4.3, I apply all the previously presented solvers to a medium with a complex velocity model and a complex topography.

In sections 4.1 and 4.2, the numerical illustrations are performed in four isotropic models, defined inside a 4×4 km square.

- **Homogeneous velocity**: The first model consists of a homogeneous velocity $c = 2000$ m/s.
- **Vertical gradient of velocity**: In the second model, the velocity varies linearly from 2000 m/s at the surface to 10000 m/s at maximum depth.
- **Gaussian velocity perturbations**: In a homogeneous background model with a velocity equals to 2000 m/s, I add two Gaussian velocity perturbations. One is positive, the other one is negative.
- **Velocity layers**: I consider two layers. The velocity inside the top layer, where the source is located, is equal to 2000 m/s. The velocity in the bottom layer is 2300 m/s. A small smooth transition layer of 30 m width is added between the layers.

These models are displayed in figure 4.1. In all the experiments, the point-source is located at $x_s = 2$ km, $z_s = 2$ km.



(a) Homogeneous velocity

(b) Vertical gradient of velocity

(c) Gaussian velocity perturbations

(d) Velocity layers

Figure 4.1: The four models used in the numerical experiments in sections 4.1 and 4.2.

4.1 Solving Hamilton–Jacobi equation for take-off angle

In this section, I consider the equation verified by the take-off angle in two dimensions. This equation could be solved with an FD approach (Belayouni, 2013). However, this equation is a Hamilton–Jacobi equation. Therefore, I choose to solve it using the same FSM-DG scheme as for traveltime computation (see chapter 3). In the following, I explain how to adapt the scheme to this equation, and I give some numerical illustrations.

At every point $\boldsymbol{x}$ of the medium (in particular, at any receiver $\boldsymbol{r}$), the take-off angle $\phi_{\boldsymbol{s}}(\boldsymbol{x})$ is the shooting angle, at the source $\boldsymbol{s}$, which yields the ray linking the source $\boldsymbol{s}$ and the considered point $\boldsymbol{x}$. The take-off angle equation derived in section 1.1.2.3 in an isotropic medium expresses the conservation of the take-off angle along a ray. It is written as

$$\nabla_{\boldsymbol{x}}\phi_{\boldsymbol{s}}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}} T(\boldsymbol{x}) = 0, \tag{4.1}$$

with a suitable point-source condition in the vicinity of the source $\boldsymbol{s}$, that could be written

$$\phi_{\boldsymbol{s}}(r, \theta) = \theta, \tag{4.2}$$

in polar coordinates (r, θ), for an arbitrarily small $r \neq 0$.

In my work, I have developed the numerical solver for the take-off angle in 2D isotropic media only. However, an equation for the anisotropic case is given by (1.36), and the extension to three dimensions is straightforward and requires to solve two equations similar to equation (4.1), independently, instead of one.

4.1.1 Numerical method

4.1.1.1 Factorization

In order to increase the accuracy of the computation, the factorization technique can be employed again. The take-off angle $\phi_{\boldsymbol{s}}$ is written as the sum of a reference solution ϕ_0 and a perturbation φ (I ommit the $\boldsymbol{x}$ and $\boldsymbol{s}$ subscripts in the following for simplicity of notations). Thus, I can write equation (4.1) as

$$\nabla(\phi_0 + \varphi) \cdot \nabla T = 0. \tag{4.3}$$

Recalling the factorization of the Eikonal equation from section 2.1.2.1, I obtain

$$\nabla(\phi_0 + \varphi) \cdot \nabla(T_0 + \tau) = 0, \tag{4.4}$$

which develops like

$$\nabla\phi_0 \cdot \nabla T_0 + \nabla\phi_0 \cdot \nabla\tau + \nabla\varphi \cdot \nabla T_0 + \nabla\varphi \cdot \nabla\tau = 0. \tag{4.5}$$

The reference solution ϕ_0 is chosen in the same medium as the reference solution for traveltime: a homogeneous medium, where the rays are straight lines emerging from the source, so that the exact angle is known in the whole medium. In the homogeneous reference medium, equation (4.1) is verified for quantities ϕ_0 and T_0. Therefore, the first term in equation (4.5) simplifies, yielding

$$\nabla\phi_0 \cdot \nabla\tau + \nabla\varphi \cdot \nabla T = 0. \tag{4.6}$$

A similar factorization for the take-off angle computation is performed in Belayouni (2013).

4.1.1.2 FSM-DG scheme

Equation (4.6) is a linear HJ equation, unlike the Eikonal equation. I use the same FSM-DG scheme as in the case of Eikonal to solve it, keeping the same discretization of the medium. The pseudo-time formulation for $\varphi(\boldsymbol{x}, \xi)$ now writes

$$\partial_\xi\varphi + \nabla\phi_0 \cdot \nabla\tau + \nabla\varphi \cdot \nabla T = 0. \tag{4.7}$$

It is then possible to apply the DG scheme (2.39) with the Hamiltonian

$$\mathcal{H}(\boldsymbol{x}, \nabla\varphi) = \nabla\phi_0 \cdot \nabla\tau + \nabla\varphi \cdot \nabla T. \tag{4.8}$$

We immediately notice that the derivatives of the Hamiltonian with respect to the spatial derivatives of the unknown φ are equal to the traveltime gradient:

$$\nabla_{\nabla\varphi}\mathcal{H} = \nabla T. \tag{4.9}$$

It means that the take-off angle equation has the same characteristics as the underlying Eikonal equation. The Roe speed has the same orientation as in the Eikonal case, so that in the DG scheme, the causality is governed by the same quantity. I can take benefit from this for the numerical computation: quantities like ∇T and its approximation at interfaces between elements are involved for the traveltime computation, and can be reused as such for the take-off angle computation, at the same discrete points, since I use the same discretization. The first-order derivatives of T are involved in the take-off angle equation, but no higher-order derivatives. Therefore, a prior P^1 computation of the traveltime is enough to compute the take-off angle.

The boundary conditions enforced as an additional term inside scheme (2.39) might be kept as such for the take-off angle. However, I emphasize that this term always equals to 0, since the boundary conditions are already enforced when computing traveltime by solving Eikonal. This comes from the fact that $\mathcal{H}^-_{\boldsymbol{n}_{K_i}}(\boldsymbol{x}, \xi)$ is always positive.

4.1.2 Numerical illustrations

In this section, I show the take-off angle field computed in the four reference models (figure 4.1). The mesh is Cartesian, with P^1 rectangular elements and $N_x = N_z = 200$. In the case of a gradient of velocity, I compare the numerical solution with an exact solution to exhibit the accuracy of the DG scheme for the take-off angle computation.

Homogeneous velocity In a homogeneous medium, the rays are straight. This is shown in figure 4.2 where the isocontours of the take-off angle field, corresponding to rays, are indeed straight lines emerging from the source point.

Vertical gradient of velocity Results are shown in figure 4.3. The isocontours of the take-off angle field, corresponding to rays, are portions of circles. This is what is expected in a model with a gradient of velocity. A closed-form solution for the take-off angle in presence of a constant vertical gradient of velocity Γ is given in Virieux (1996, eq. 6.108) and I reproduce it here. The medium velocity is given by

$$c(z) = c_0 + (z - z_0)\,\Gamma. \tag{4.10}$$

Consider a ray shot from a source point located at ($z = z_0$, $x = x_0$) with an angle θ_0 to the vertical axis. At a given depth $z^\star$, the lateral position $x(\theta_0)$ is given by

$$x - x_0 = \frac{c_0}{\Gamma \sin\theta_0}\left(\cos\theta_0 - \sqrt{1 - \frac{c^2(z^\star)}{c_0^2}\sin^2\theta_0}\right). \tag{4.11}$$

I perform the numerical comparison in the following way: I successively consider two depths $z^\star = 2500$ m and $z^\star = 3800$ m in the model. I define a set of θ_0, and compute the corresponding $x(\theta_0)$ using expression (4.11). I finally compare the values of θ_0 with the computed values extracted at the same points from the Eulerian solution obtained by the DG scheme when solving equation (4.1). The results are shown in figure 4.4. The DG values exhibit an excellent fit with the exact ones. The errors of the computed values have an order of magnitude of 10^{-4}. This validates the accuracy of this numerical approach.

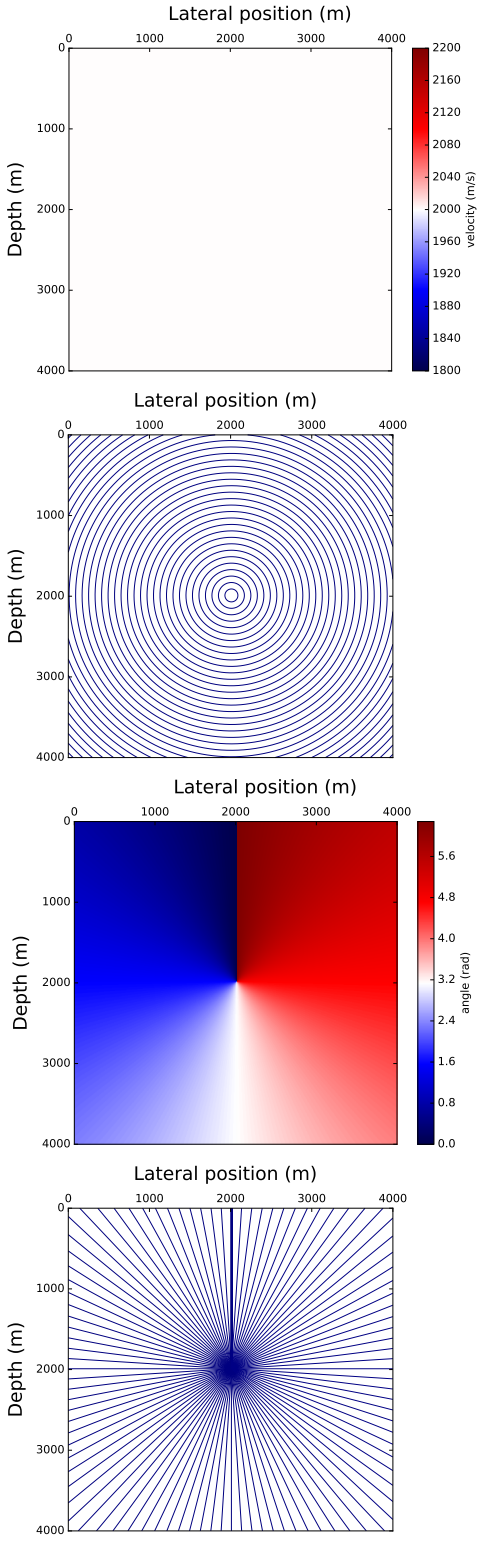


Figure 4.2: Homogeneous velocity. From top to bottom: Velocity model; isocontours of traveltime; take-off angle field; isocontours of the take-off angle field (akin to rays).

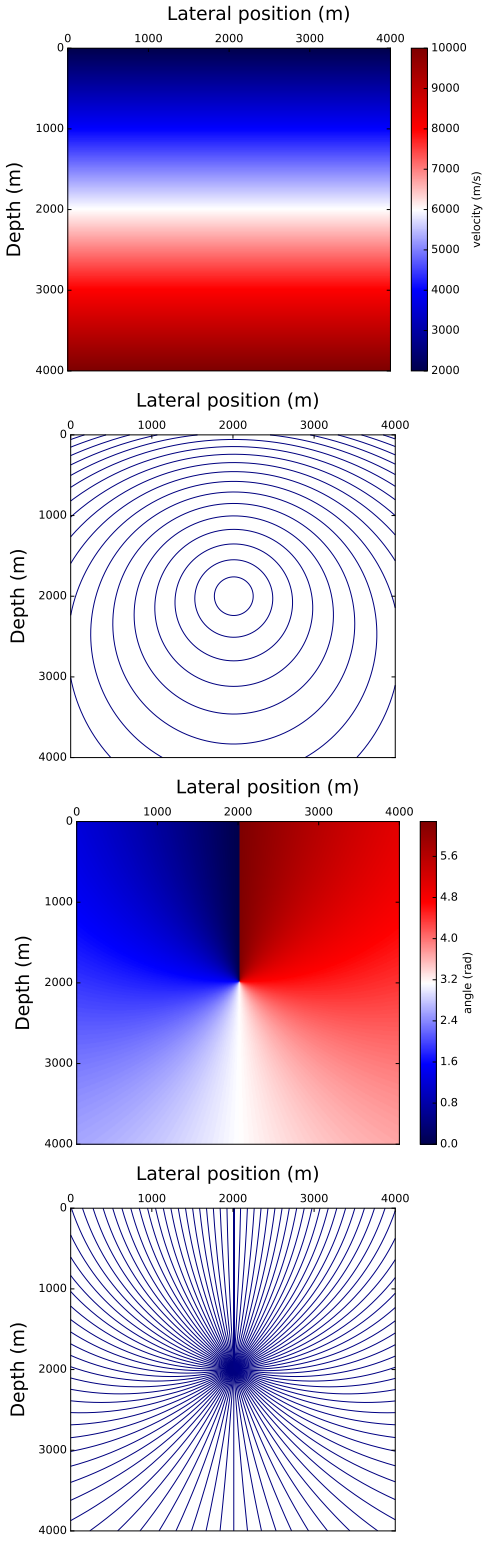


Figure 4.3: Vertical gradient of velocity. From top to bottom: Velocity model; isocontours of traveltime; take-off angle field; isocontours of take-off angle field (akin to rays).

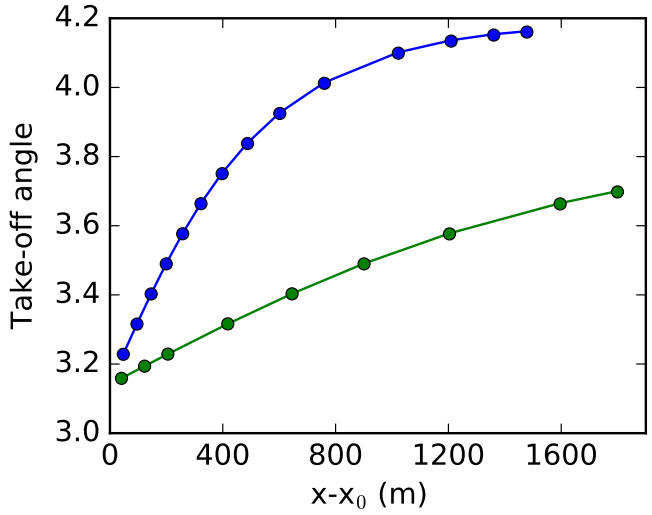


Figure 4.4: Take-off angle in the medium with a velocity gradient, at depth $z^\star = 2500$ m (blue) and $z^\star = 3800$ m (green). The plain lines correspond to the exact solution, while the values extracted from the DG solver are plotted with circles. The errors of the DG numerical values have an order of magnitude of 10^{-4}.

Gaussian velocity perturbations Results obtain in this model are shown in figure 4.5. Notice the converging rays near the negative anomaly, and the diverging rays near the positive one.

Velocity layers Results are shown in figure 4.6. The head wave comes from one unique ray with critical incident angle at the interface between the two layers, so that the take-off angle in this region is constant.

4.1.3 Conclusion

I have shown that the FSM-DG scheme, initially designed for traveltime computation, can also be used for the computation of the take-off angle. Causality and boundary conditions are automatically handled from the prior traveltime computation.

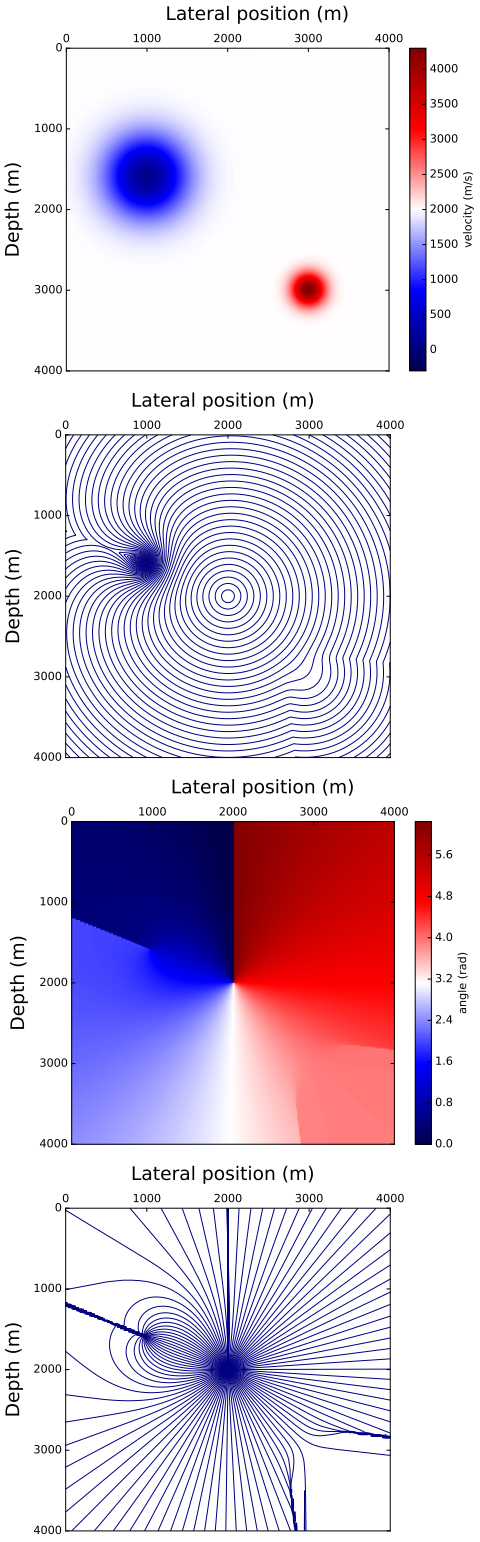


Figure 4.5: Gaussian velocity perturbations. From top to bottom: Velocity model; isocontours of traveltime; take-off angle field (akin to rays); isocontours of take-off angle field (akin to rays).

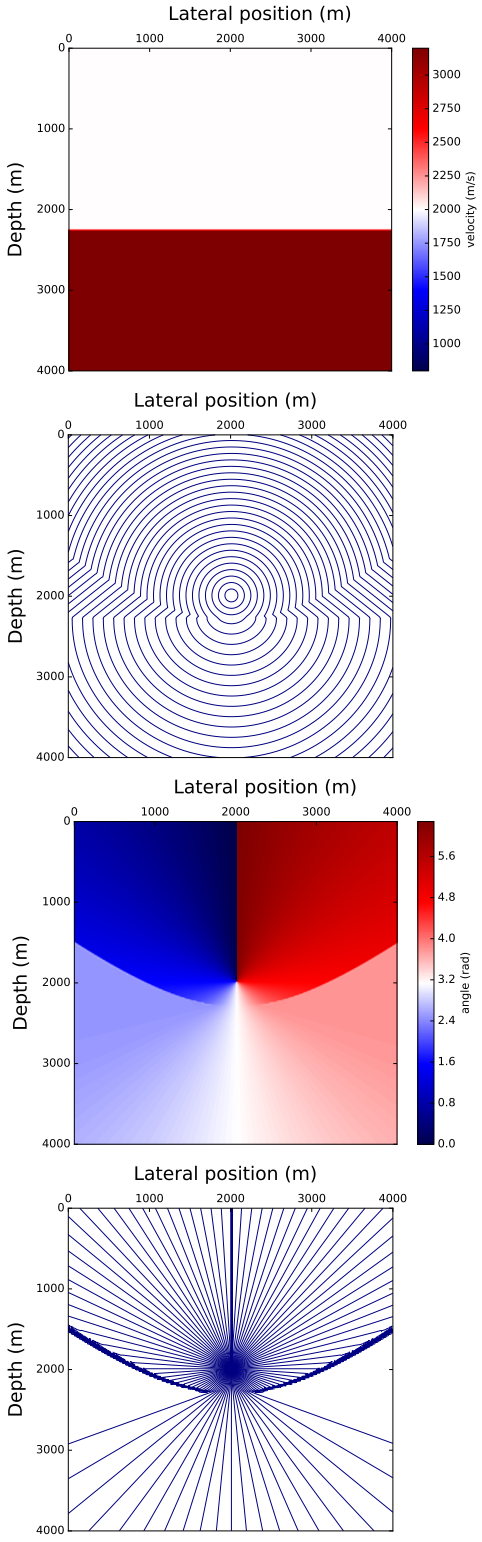


Figure 4.6: Velocity layers. From top to bottom: Velocity model; isocontours of traveltime; take-off angle field; isocontours of take-off angle field (akin to rays).

4.2 Solving transport equation for amplitude and adjoint-state variable

In this section, I consider the transport equation, verified by the amplitude as well as the adjoint-state variable in the case of traveltime tomography. I derive a finite-volume (FV) solver for the transport equation, and I exhibit some numerical illustrations. I have designed algorithms and performed illustrations in 2D only; however the extension to 3D would need only few modifications and should be straightforward.

4.2.1 Amplitude computation

4.2.1.1 Introduction

In high-frequency wave propagation, the transport equation expresses an energy conservation. The energy flux is carried by the group velocity vector $\vec{\mathcal{U}}$, and the general transport equation writes in a conservative form

$$\nabla \cdot (a\vec{\mathcal{U}}) = 0, \tag{4.12}$$

where the quantity a is conveyed along $\vec{\mathcal{U}}$. Suitable boundary conditions must be added. In all the following I consider an omnidirectional point source. The solution in the vicinity of the source can be derived in a closed form by assuming a homogeneous velocity in this vicinity. This is performed in Červený (2001, eq. 2.5.10) and Virieux (1996, eq. 6.72).

The transport equation can be solved with an FD approach (Buske and Kästner, 2004; Belayouni, 2013). Instead, I consider a DG formulation and a lower-order FV Eikonal solver, keeping a similar discretization for traveltime and amplitude.

Since $\vec{\mathcal{U}}$ depends on the components of ∇T, equation (4.12) involves second-order derivatives of the traveltime. Interestingly, this formulation can be integrated to involve only first-order derivatives of the traveltime. This can be done in two different ways:

- **DG formulation**: the transport equation can be multiplied by a test function and integrated by parts, yielding a volume integral involving the amplitude, the first-order derivatives of the traveltime, and the first-order derivatives of the test function. A flux term naturally appears as a surface integral at the element interfaces.
- **FV formulation**: the Green–Ostrogradsky formula locally transforms the conservative form of the transport equation into a flux integral through the element boundaries.

Formally, both formulations are strongly related, as the DG case can be viewed as a higher-order generalization of the FV case (Cockburn, 2003). Here, in both cases, the causality might be enforced by carefully writing the flux terms in an upwind way. This amounts to respect the direction of propagation provided by the Roe speed at interfaces and coming from the Eikonal solution. I have tried both DG and FV methods. Both yield similar results in smooth regions of the solution. However, near the singularities, the DG method yields spurious oscillations. To get rid of them, it is possible to implement a slope limiter, as a post-treatment. Another way is to take the mean of the value inside each element as the solution. However, I have found that the most simple solution is to consider the FV approach, which is faster and eventually yields the same results, with no oscillations, since the method is lower-order. In some way, the FV approach is the most efficient way to retrieve a mean of the solution in the elements.

The physical meaning of a in equation (4.12) depends on the underlying wave equation considered, which might be scalar, acoustic, elastic, etc. Several expressions for a in the transport equation are given in Červený (2001, pp. 63–71). In the following, I focus on the isotropic case, where the group velocity is orthogonal to the wavefront, therefore orthogonal to ∇T. I have not developed the numerical algorithm for the anisotropic case. In theory it could be derived in a straightforward manner from the isotropic case, although in the preliminary tests that I have performed, I observe artifacts and instabilities that should be further investigated. The isotropic transport equation writes

$$\nabla \cdot (a \nabla T) = 0. \tag{4.13}$$

Next, I describe how to build a finite-volume solver for this transport equation.

4.2.1.2 A finite-volume solver for the amplitude

FV local upwind scheme I keep the same mesh as for the traveltime computation. However, the solution is no longer described as a polynomial over each element, but as a single value a_i per element K_i, which can be viewed as the mean value of the solution for each element. This corresponds to a P^0 approximation. I define $\boldsymbol{n}_{K_i}$ to be the outward unit normal to the boundary of element K_i. The local FV scheme for solving equation (4.13) writes

$$\int_{\partial K_i} a^\star \nabla T \cdot \boldsymbol{n}_{K_i} \,\mathrm{ds} = 0, \tag{4.14}$$

where $a^\star$ must be chosen at the element interface. Considering an interface of element K_i, I denote by a^- the value of a inside the element, so that $a^- = a_i$. I denote by a^+ the value of a inside the neighboring element. Therefore, an upwind formulation of the FV scheme consists of choosing $a^\star$ depending on the orientation of ∇T inward or outward as follows:

$$a^\star \nabla T \cdot \boldsymbol{n}_{K_i} = a^- \max(\nabla T \cdot \boldsymbol{n}_{K_i}, 0) + a^+ \min(\nabla T \cdot \boldsymbol{n}_{K_i}, 0), \tag{4.15}$$

yielding the upwind local FV scheme

$$\int_{\partial K_i} \big(a^- \max(\nabla T \cdot \boldsymbol{n}_{K_i}, 0) + a^+ \min(\nabla T \cdot \boldsymbol{n}_{K_i}, 0)\big) \,\mathrm{ds} = 0. \tag{4.16}$$

The values of ∇T at interfaces are extracted from the prior traveltime computation. Since only the first-order derivatives of the traveltime are involved, a P^1 computation of the traveltime is enough.

Point-source factorization Here again, it is possible to implement the point-source factorization, in order to implement the source condition and increase the accuracy of the solver. Again, I consider a reference medium where the amplitude is known, so that a reference solution a_0 is defined. I then write the additive factorization as

$$a = a_0 + l, \tag{4.17}$$

where l is the perturbation field for the amplitude. In an isotropic homogeneous medium with velocity c_0, I use the following solution (from e.g. Luo et al., 2014a) which writes in the 2D case

$$a_0(\boldsymbol{x}; \boldsymbol{x}_s) = \frac{c_0}{8\pi |\boldsymbol{x} - \boldsymbol{x}_s|} \tag{4.18}$$

for an omnidirectional point source located at $\boldsymbol{x}_s$. In practice, this solution is enforced inside a few elements all around the source instead of only in the source element, otherwise the accuracy deteriorates.

External boundary conditions At the interface of an element located along a boundary of the domain, only a^- is defined, since there is no neighbor outside of the domain. Hence, scheme (4.16) might be undefined. However, the transport vector ∇T is oriented outwards. This is enforced in the Eikonal DG solver. Therefore, in the upwind scheme (4.16), only the term involving a^- will have a non-zero value. In other words, only this term is kept in the scheme at domain boundaries.

Local and global solvers Locally, I use again a pseudo-time integration with an explicit Euler step to reach the static solution. The solution is reached in very few local iterations. To solve the problem globally, I use again the FSM solver in the same way as for the traveltime, updating elements one after one respecting the global orderings.

4.2.1.3 Numerical illustrations

In the numerical illustrations, I consider the scalar wave equation, so that a corresponds to the amplitude squared of the high-frequency solution of the scalar wave equation. In that case, the amplitude variation in space is related to the geometrical spreading, expressing how the energy flux, carried by the group velocity vector, diverge or converge, but also by the variation of speed in the medium (Virieux, 1996, eq. 6.72). Interestingly, both effects can be separated. Writing $a(\boldsymbol{x}) = b(\boldsymbol{x})c(\boldsymbol{x})$, we obtain an alternative transport equation

$$\nabla \cdot (b(\boldsymbol{x})c(\boldsymbol{x})\nabla T) = 0, \tag{4.19}$$

where $c(\boldsymbol{x})$ is the velocity field, so that the unknown $b(\boldsymbol{x})$ corresponds to the inverse of the geometrical spreading. In the following examples, I exhibit results for both equations (4.19) for $b(\boldsymbol{x})$ (geometrical spreading) and (4.13) for $a(\boldsymbol{x})$ (amplitude squared), showing the geometrical spreading effect and the total amplitude shape including the velocity compensation, respectively.

Next, I show the amplitude field computed in the four reference models (figure 4.1). The mesh is Cartesian, with P^1 (respectively P^0) rectangular elements for traveltime (respectively amplitude) computation and $N_x = N_z = 200$. In the case of a gradient of velocity, I compare the numerical solution with an exact solution to exhibit the accuracy of the FV scheme.

Homogeneous velocity In a homogeneous medium, the seismic amplitude decay is proportional to $1/\sqrt{|\boldsymbol{x} - \boldsymbol{x}_s|}$. Results are shown in figure 4.7. The factorization with the exact solution yields a nearly exact numerical solution. Since the velocity is homogeneous, the amplitude variation is only affected by the geometrical spreading, so that both results obtained from equations (4.19) and (4.13) are the same.

Vertical gradient of velocity Results are shown in figure 4.8. First, some grid artifacts appears along the vertical axis on the amplitude isocontours. These artifacts are linked to the mesh geometry, and I have not investigated further on them so far. Second, in the bottom part of the domain, the characteristics of the Eikonal (rays) diverge. The geometrical spreading in this region is high. On the contrary, the geometrical spreading is low in the upper part of the domain. Therefore, a faster decrease of the amplitude is observed when solving equation (4.19) in the bottom part than in the upper part of the domain. However, this effect is compensated by the velocity gradient. After the velocity compensation, the total amplitude from equation (4.13) exhibits an opposite behavior. This interesting example shows that the transport equation must be considered with care. The interpretation of the results depend on

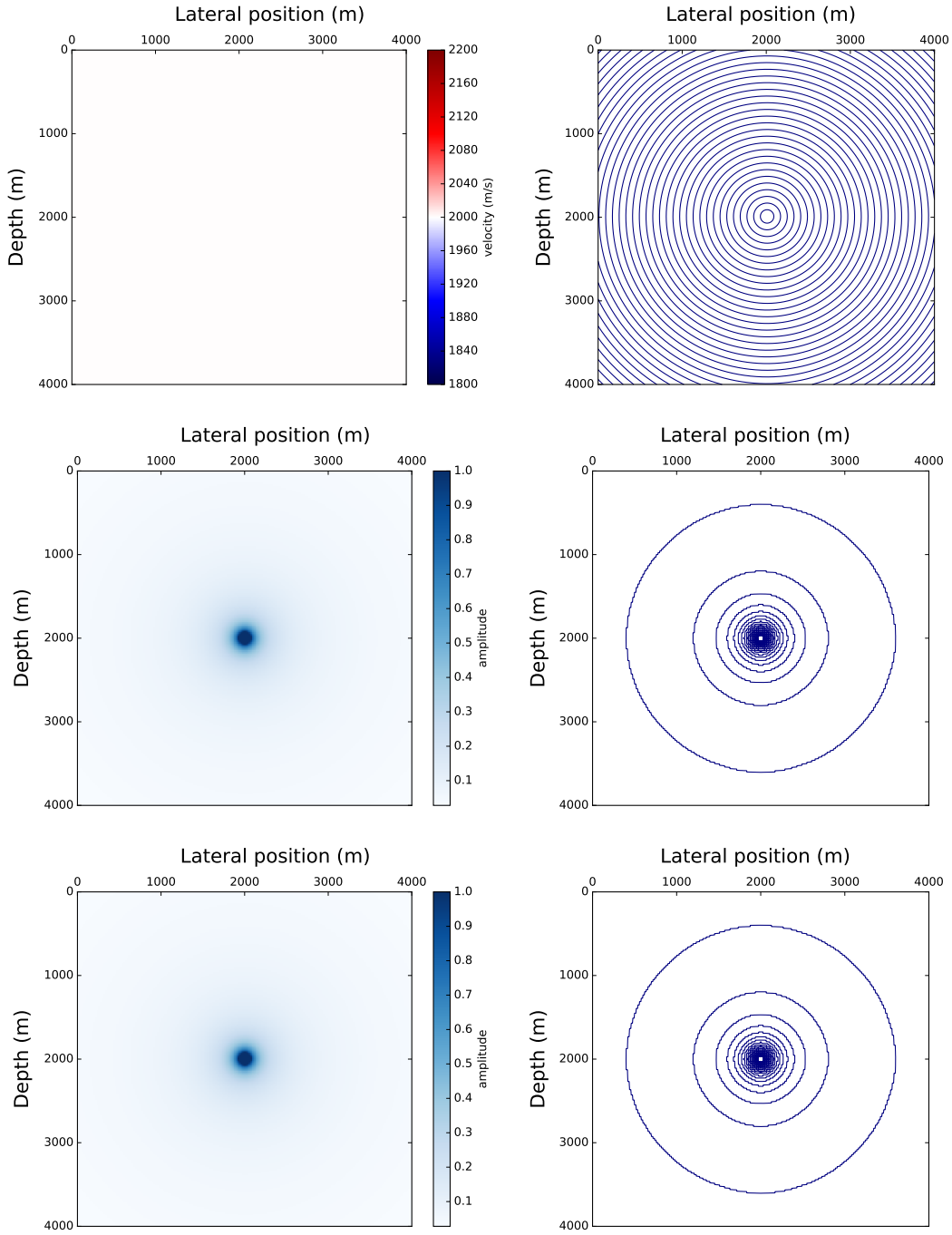


Figure 4.7: Homogeneous velocity. Top left: Velocity model; top right: isocontours of traveltime; center: solution of equation (4.19) (inverse of the geometrical spreading) and its isocontours; bottom: solution of equation (4.13) (amplitude squared) and its isocontours. Since the velocity is homogeneous, solutions of equations (4.19) and (4.13) are the same.

the underlying equation. For instance, in an elastic case, the interpretation of the transport equation for solving the amplitude of displacement fields would be quite different.

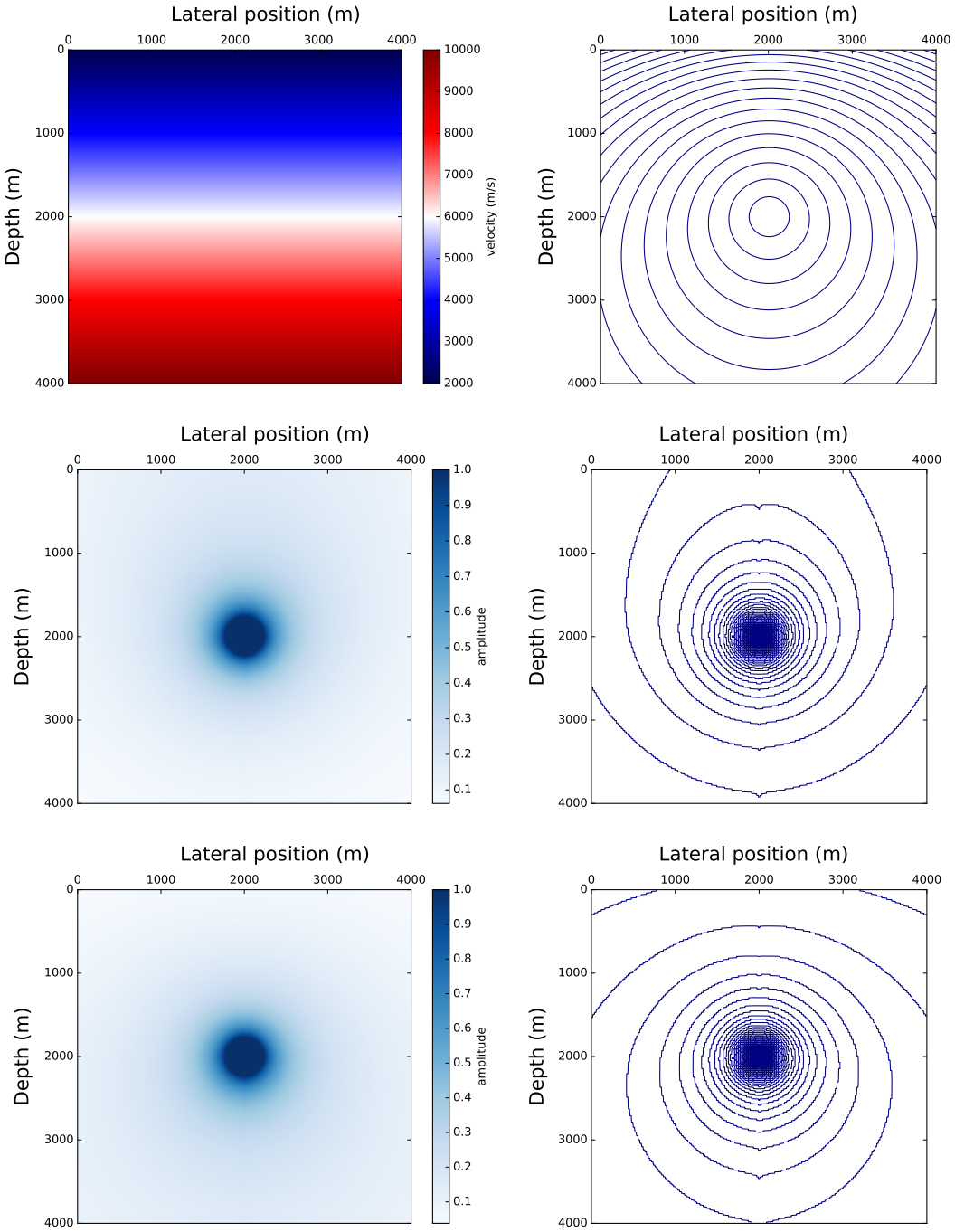


Figure 4.8: Vertical gradient of velocity. Top left: Velocity model; top right: isocontours of traveltime; center: solution of equation (4.19) (inverse of the geometrical spreading) and its isocontours; bottom: solution of equation (4.13) (amplitude squared) and its isocontours. The ray divergence in the bottom part of the model (top right plot) yields a high geometrical spreading, thus a faster decreasing of the amplitude than in the top part of the model (center). However, this effect is compensated by the high velocity in this area. After the velocity compensation, the total amplitude exhibits an opposite behavior (bottom).

In a model with a gradient of velocity Γ, it is possible to derive a closed-form solution for the amplitude. This solution is given in Virieux (1996, eq. 6.109) and I reproduce it here. The medium velocity is given by

$$c(z) = c_0 + \Gamma(z - z_0). \tag{4.20}$$

Consider a ray shot from a source point located at $z = z_0$, $x = x_0$ with an angle θ_0 to the vertical axis. At a given depth $z^\star$, the lateral position $x(\theta_0)$ is given by

$$x - x_0 = \frac{c_0}{\Gamma \sin\theta_0} \left(\cos\theta_0 - \sqrt{1 - \frac{c^2(z^\star)}{c_0^2} \sin^2\theta_0} \right). \tag{4.21}$$

The geometrical spreading is expressed by the Jacobian J, which can be calculated using the identity

$$J(\theta) = \left| \frac{\partial x}{\partial \theta_0} \right|_{z^\star} \sqrt{1 - \frac{c^2(z^\star)}{c_0^2} \sin^2\theta_0}, \tag{4.22}$$

where the derivative of x with respect to θ_0 is obtained from expression (4.21). The numerical comparison is performed as follows: I successively consider two depths $z^\star = 2500$ m and $z^\star = 3800$ m in the model. I first define a set of θ_0, and compute the corresponding $x(\theta_0)$ using expression (4.21). Next, I obtain the values of J at points $z^\star, x(\theta_0)$ for the set of θ_0 using expression (4.22). I finally compare these exact solutions with the values extracted at the same points from the Eulerian solution obtained by the FV scheme when solving equation (4.19). More precisely, I compare the FV solution and the inverse of the Jacobian $1/J$. Indeed, the geometrical spreading is easier to understand as the ratio of the Jacobian between two points. I thus define an absolute reference point at $x = 2500$ m, $z = 2500$ m where I arbitrarily set $J = 1$. The results are shown in figure 4.9. The FV values exhibit a very good fit with the exact ones. This validates the numerical approach.

Gaussian velocity perturbations Results are shown in figure 4.10. Notice the high-amplitude lines forming along the singularities, where two branches of the solution converge. These lines are due to the discrete representation of line singularities, which leads to a merging of the two branches. Fortunately, this non-physical merging is constricted to an area along the singularity, the width of which is linked to the mesh size: a finer mesh leads to a narrower line artifact.

Velocity layers Results are shown in figure 4.11. The head wave comes from one unique ray with critical incident angle at the interface between the two layers, so that its theoretical amplitude is zero. We nicely retrieve this property numerically.

4.2.1.4 Conclusion

In conclusion, it is possible to solve the transport equation for the amplitude in an FSM-FV approach. Some numerical artifacts appear, and more developments should be performed in order to try to get rid of them. However, in the applications I consider later and in most of the realistic applications, the accuracy needed for the amplitude is far less critical than for the traveltime.

Here, the transport equation is solved in a pure 2D configuration. This means that the propagation is considered to happen only in a 2D plane, or equivalently that a line source is defined in a 3D model with constant properties along one dimension. For real 3D data experiments with point sources, the

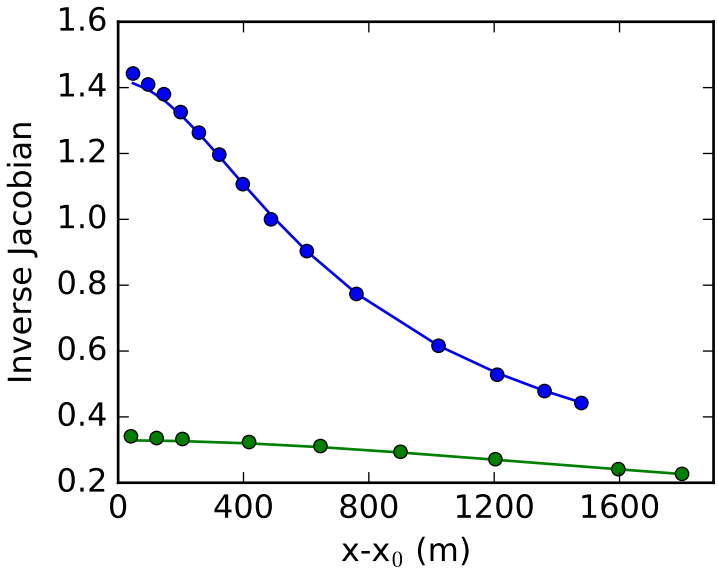


Figure 4.9: Inverse of the Jacobian in the medium with a velocity gradient, at depth $z^\star = 2500$ m (blue) and $z^\star = 3800$ m (green). The plain lines correspond to the exact solution, while the values extracted from the FV solver are plotted with circles.

algorithm should be extended to 3D, or at least 2.5D if small variations in one direction are considered. The 2.5D case requires some more computational effort, since we have to account for an out-of-plane spreading factor, verifying another advection equation (Qian and Symes, 2002a; Luo and Qian, 2011).

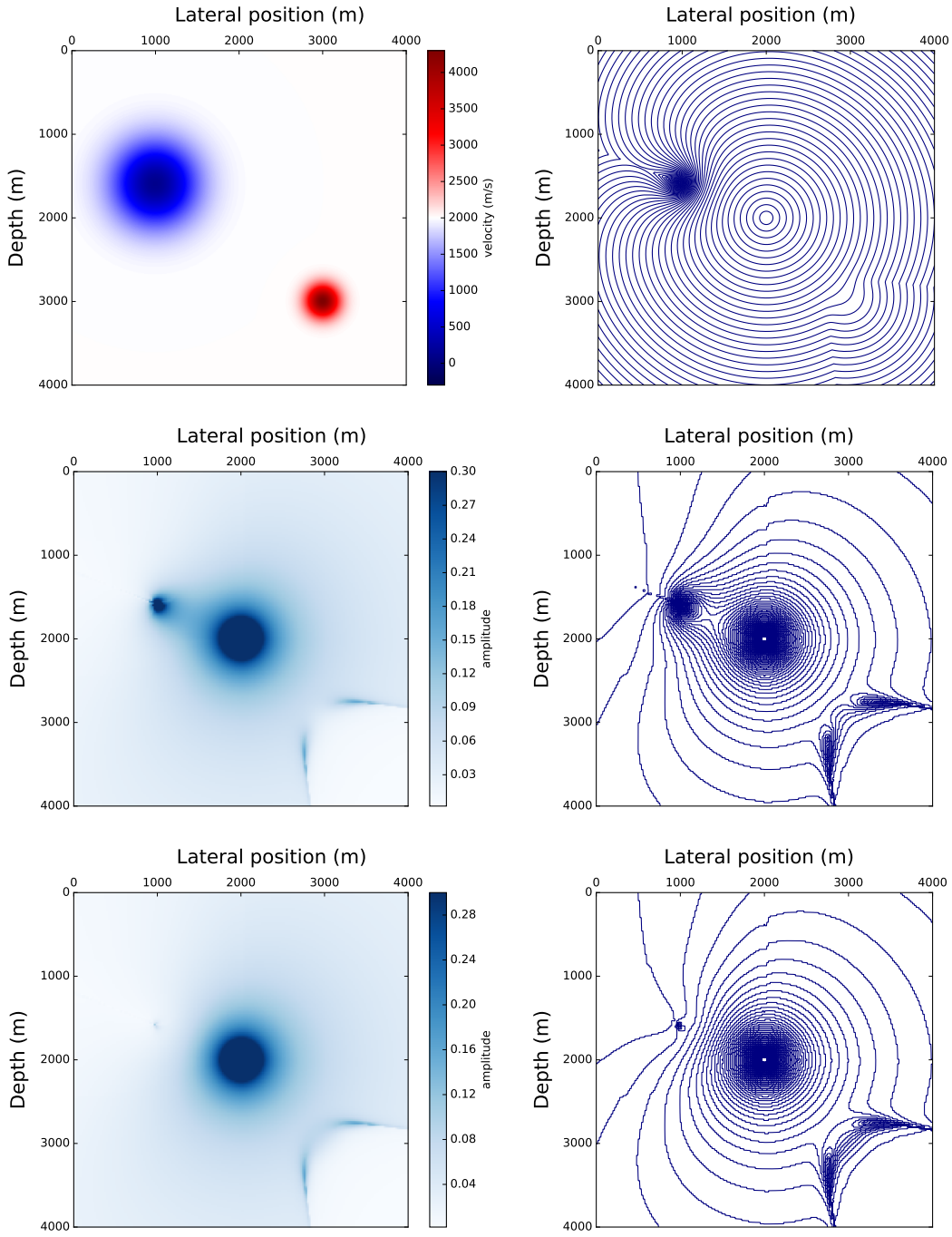


Figure 4.10: Gaussian velocity perturbations. Top left: Velocity model; top right: isocontours of traveltime; center: solution of equation (4.19) (inverse of the geometrical spreading) and its isocontours; bottom: solution of equation (4.13) (amplitude squared) and its isocontours.

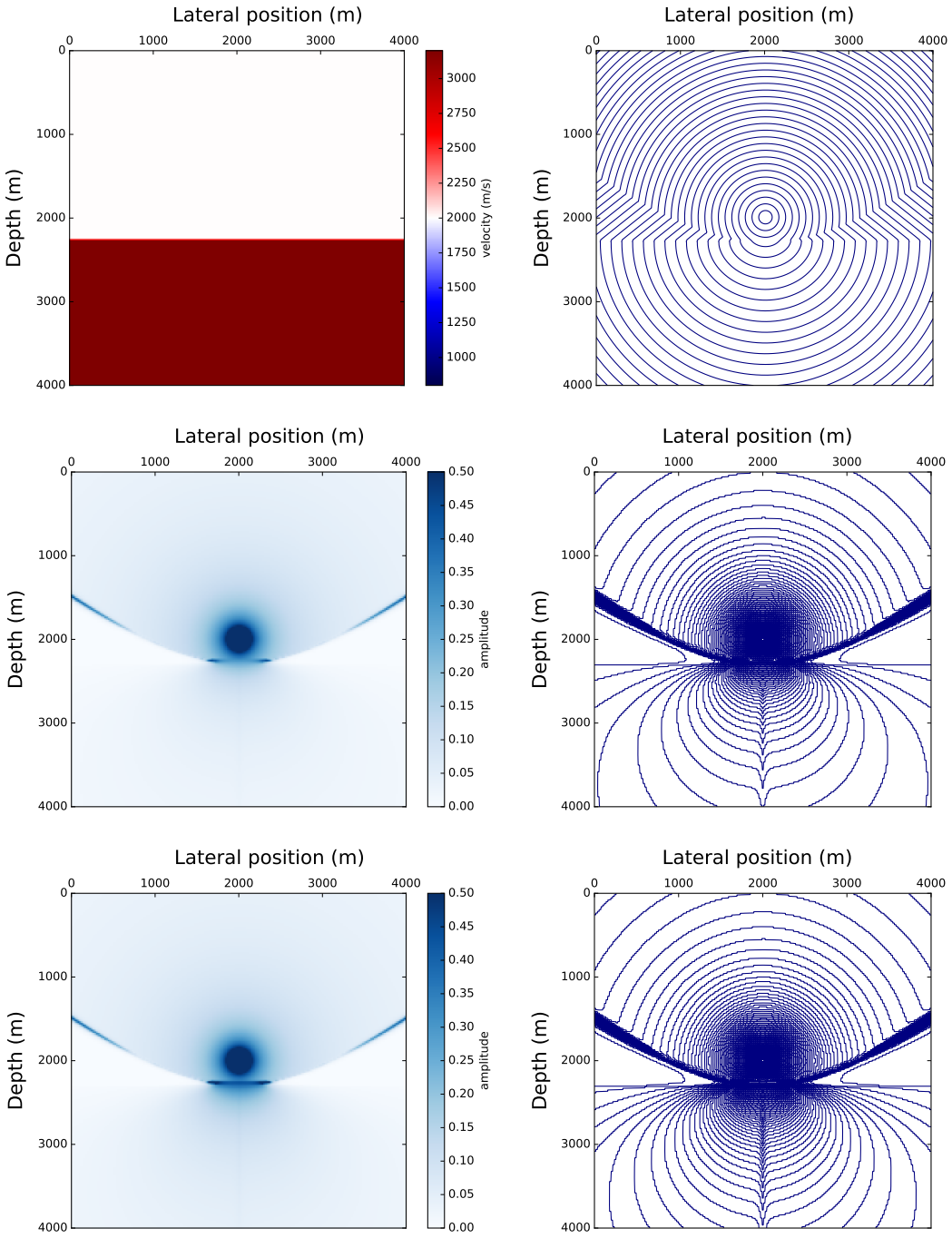


Figure 4.11: Velocity layers. Top left: Velocity model; top right: isocontours of traveltime; center: solution of equation (4.19) (inverse of the geometrical spreading) and its isocontours; bottom: solution of equation (4.13) (amplitude squared) and its isocontours. Notice the null amplitude of the head wave.

4.2.2 Adjoint-state variable computation

4.2.2.1 Introduction

Traveltime tomography is a well established, widespread tool for retrieving the long-wavelength components of the velocity in the Earth and imaging subsurface structure at all scales. Based on the principles of inverse theory (Menke, 1984; Tarantola, 2005), it includes four steps which are: model parameterization, forward calculation, inversion, and analysis of solution robustness (Rawlinson and Sambridge, 2003). The inversion step is often performed using a gradient method. Its principle is to minimize an objective function, that is, solving the least-squares problem for traveltime. The minimization involves the computation of the partial derivatives of the traveltime with respect to the model parameters, referred to as the Fréchet derivatives. The corresponding Fréchet matrix might be huge when the parameterization is fine. For one source, denoting by n the number of data available (receivers) and by m the number of model parameters, the Fréchet derivatives has an $n \times m$ complexity. Alternatively, the adjoint-state method allows to compute a gradient for the model update in a very efficient way by incorporating the residuals at receivers into a single PDE solving step for each source, so that the resulting complexity is only m. The PDE to be solved is a transport equation (Plessix, 2006; Fichtner et al., 2006). Denoting the adjoint-state variable by λ, this equation writes in the general form

$$\nabla \cdot (\lambda \vec{u}) = \mathcal{R}. \tag{4.23}$$

Here, $\mathcal{R}(\boldsymbol{x})$ denotes an appropriate source term in the right-hand side of the equation corresponding to the traveltime residuals introduced at the receiver positions. The vector $\vec{u}$ denotes either $-\nabla T$ in the isotropic case, or a quantity derived from the Hamiltonian formulation in the general anisotropic case as

$$\vec{u} = -\frac{\partial \mathcal{H}}{\partial(\nabla T)}. \tag{4.24}$$

The minus sign comes from the fact that the residuals are back-propagated from the receivers to the source. From expression (4.24) we know that $\vec{u}$ is parallel to the group velocity vector $\vec{\mathcal{U}}$, with an opposite direction. The equality $\vec{u} = -\vec{\mathcal{U}}$ is obtained for a particular Hamiltonian $\mathcal{H} = \frac{1}{2}\|\nabla T\|^2 v^2 - \frac{1}{2}$, where v is the phase velocity. For other Hamiltonians, a normalization could be performed as explained in Červený (2001, equations 4.7.3 and 4.7.4). The adjoint method relies on two steps: first, the computation of λ by solving the transport equation; second, the computation of the gradient of the cost function with respect to the model parameters. This gradient involves λ and the components of $\partial\mathcal{H}/\partial(\nabla T)$, so that the normalization is not needed as long as the same Hamiltonian is used for the two steps.

The adjoint-state approach has been implemented by Luo et al. (2012); Taillandier et al. (2009) for traveltime tomography, and an extension to anisotropic media was proposed by Waheed et al. (2016). A slope tomography using the adjoint-state method was proposed by Tavakoli F. et al. (2017); Sambolian et al. (2018).

Similar to the amplitude, the adjoint-state variable for (slope) tomography relies on a transport equation. This is the reason why I study this adjoint-state variable computation in my work, and I present here the results that can be obtained with an FV solver. This solver is currently being tested and adapted in an anisotropic tomographic formulation and should yield interesting results in the near future.

4.2.2.2 A finite-volume solver for adjoint-state variable

I use the FV solver for the amplitude with only few modifications in order to make it suitable for the adjoint-state variable computation:

- The minus sign changes the upwind scheme into a downwind one: residuals are back-propagated from receivers to the source;
- There is no boundary condition at the source. Instead, a right-hand side is added to the equation at receiver locations. Receivers are generally located inside elements; the residual is applied inside the whole volume of the element, since the FV approach corresponds to a P^0 approximation. In practice, this right-hand side is computed at the boundaries of the element as an additional flux term. When a receiver is located exactly at an interface between two elements, the residual is applied inside both elements.
- Because of the back-propagation of the residuals towards the source, the solution at the source does not affect the solution elsewhere. Therefore, there is no need for point-source factorization for the adjoint-state variable computation,

The resulting scheme can be written

$$\int_{\partial K_i} \left[\left(\lambda^- - \mathcal{R}(K_i) \right) \max(\vec{u} \cdot \boldsymbol{n}_{K_i}, 0) + \lambda^+ \min(\vec{u} \cdot \boldsymbol{n}_{K_i}, 0) \right] \mathrm{ds} = 0, \tag{4.25}$$

where $+$ and $-$ superscripts have the same meaning as in the amplitude case (expression (4.16)), and where $\mathcal{R}(K_i)$ denotes the right-hand-side corresponding to the residual of a potential receiver located inside element K_i. In the case of several receivers located inside the same element, the right-hand side contributions from each receiver might be added.

The same local and global solvers as for amplitude are employed. Since a downwind condition is enforced, there is no specific boundary condition to consider at the domain boundaries.

4.2.2.3 Numerical illustrations

For illustration purpose, I first consider the four isotropic velocity models presented in figure 4.1 and I show the resulting adjoint-state variable field obtained by solving equation (4.23) with $\vec{u} = \nabla T$. Then, I consider two simple anisotropic TTI media and I show the results obtained by solving equation (4.23) with $\vec{u} = \partial \mathcal{H} / \partial (\nabla T)$, where I use the TTI Hamiltonian from expression (1.46).

The mesh is Cartesian, with P^1 (respectively P^0) rectangular elements for traveltime (respectively adjoint-state variable) computation and $N_x = N_z = 200$. Seismic receivers are modeled all around the domain, at positions listed in table 4.1. Right-hand side values equal to 1 are introduced at each receiver position.

Homogeneous velocity In the continuum with no diffusion, the exact solution of the advection of a residual introduced at a receiver position should follow the zero-width ray between the source and the receiver. However, we have to face two numerical difficulties. First, the residual we introduce is not zero-width, due to the discretization. Second and more important, numerical diffusion occurs, which

Receiver positions		
#	x(m)	z(m)
1	0	0
2	1000	0
3	2000	0
4	3000	0
5	4000	0
6	4000	1000
7	4000	2000
8	4000	3000
9	4000	4000
10	3000	4000
11	2000	4000
12	1000	4000
13	0	4000
14	0	3000
15	0	2000
16	0	1000

Table 4.1: Receiver positions for the adjoint-state variable illustrations.

is controlled by grid effects. This yields to an enlargement of the region affected by a given residual, which depends on the mesh size and structure.

This effect is clearly illustrated in figure 4.12: the medium is homogeneous, yet the shape of the beams coming from the different receivers are not all the same. It is interesting to notice the horizontal and vertical beams, where the information propagates inside elements aligned in the grid, so that the beams are constricted into a single element width. On the contrary, the other beams are getting wider between the receiver and the source, and refocus near the source.

For tomographic applications, the resulting gradient is spatially smoothed, so that the numerical diffusion visible in figure 4.12 does not affect the results. Indeed, when considering finite-frequency effects, the underlying assumption is that the spatial discretization is small with respect to the first Fresnel zone width.

Vertical gradient of velocity Results are shown in figure 4.13. Adjoint-state variable beams are curved in the same way as seismic rays.

Gaussian velocity perturbations Results are shown in figure 4.14. The beams from receivers affected by the velocity perturbations are deformed.

Velocity layers Results are shown in figure 4.15. Notice the beams coming from the receivers located in the refracted wave: they focus along the interface between layers before joining the top layer and the source.

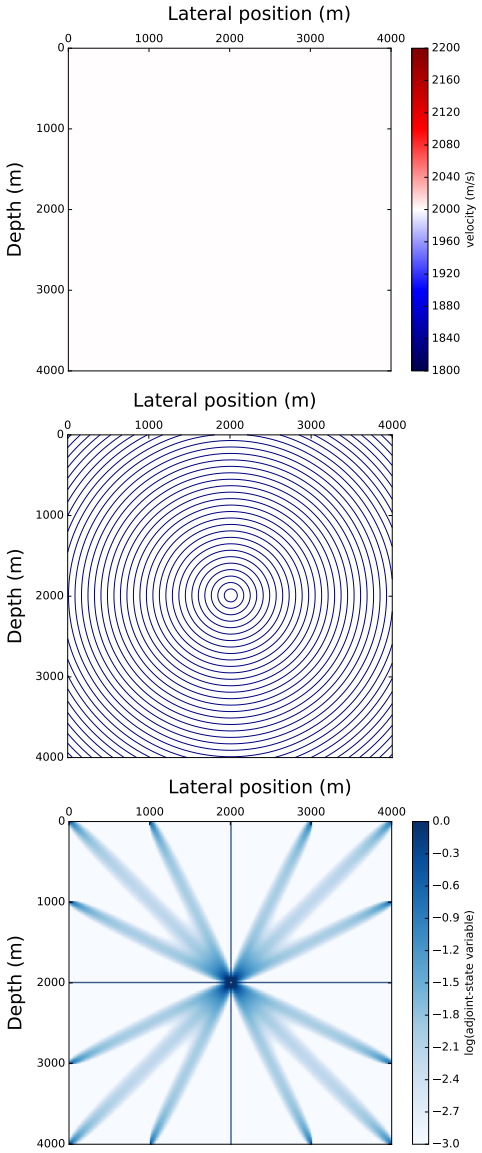


Figure 4.12: Homogeneous model. From top to bottom: Velocity model; isocontours of traveltime; logarithmic adjoint-state variable. Note the numerical diffusion due to grid effects, while the horizontal and vertical beams are not affected.

Two anisotropic media Here, I introduce homogeneous TTI anelliptic anisotropy with the following parameters:

$$\epsilon = 0.4, \quad \delta = 0.2, \quad \theta = 30 \text{ deg}. \tag{4.26}$$

In a first example, I consider a constant qP-wave vertical velocity of 2000 m/s. Results in figure 4.16 show that the adjoint-variable field is very similar to the one obtained in the isotropic constant-velocity case. This comes from the fact that rays are straight lines when the medium is homogeneous, even in presence of anisotropy.

In a second example, I consider a vertical gradient of velocity, which varies linearly from 2000 m/s at the surface to 10000 m/s at maximum depth. Results in figure 4.17 show the deformation of the beams

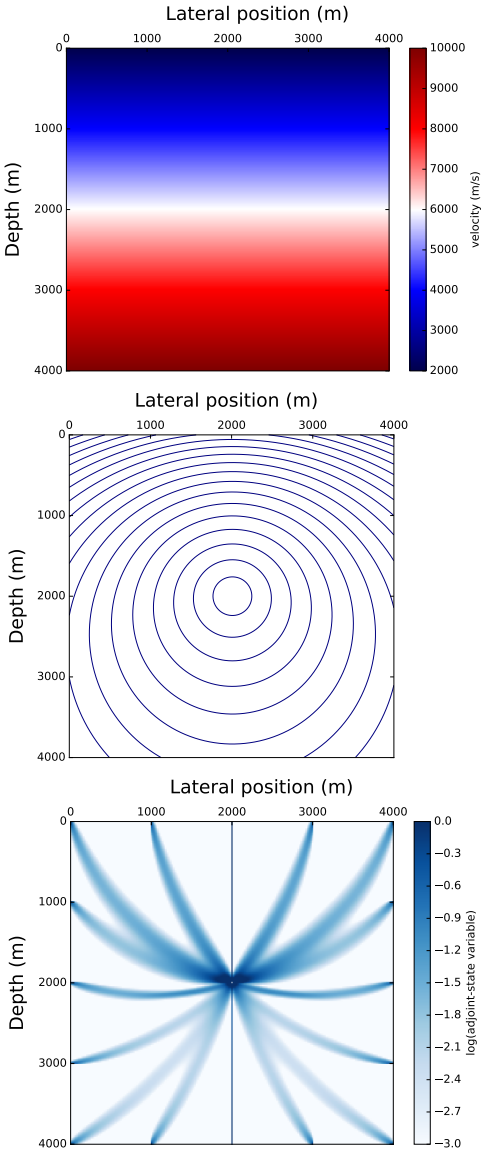


Figure 4.13: Vertical gradient of velocity. From top to bottom: Velocity model; isocontours of travel-time; logarithmic adjoint-state variable.

due to the velocity gradient. The main difference with the isotropic velocity gradient case lies in the amplitude difference between the top and bottom beams. This comes from the TTI Hamiltonian used in the adjoint-state variable computation, which is not normalized like in the isotropic case. Note that these normalization effects are compensated when the gradient is computed, as mentioned in section 4.2.2.1.

4.2.2.4 Conclusion

In conclusion, it is possible to solve the transport equation related to the adjoint-state variable using the same FSM-FV approach as for amplitude. Both isotropic and anisotropic settings are investigated.

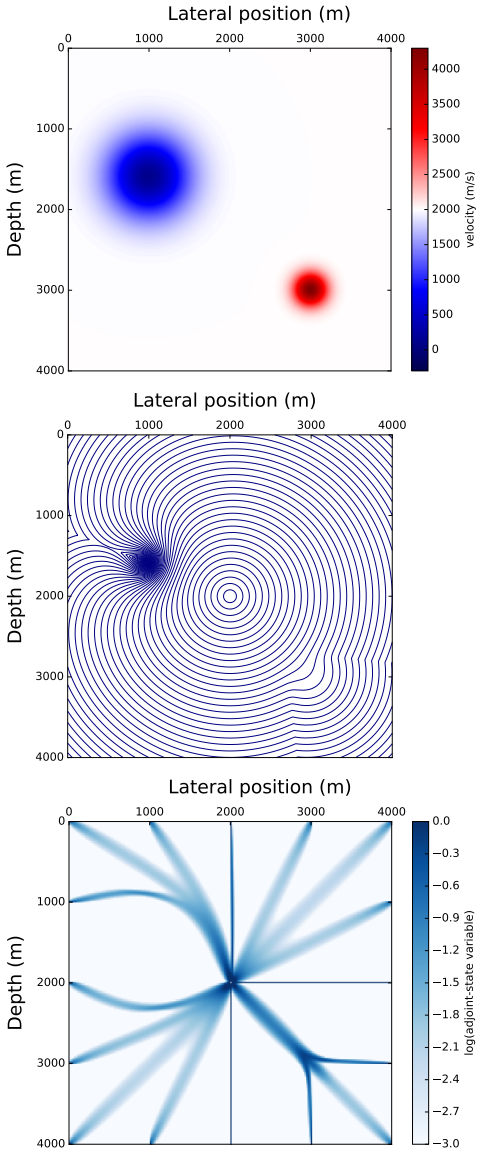


Figure 4.14: Gaussian velocity perturbations. From top to bottom: Velocity model; isocontours of traveltime; logarithmic adjoint-state variable.

Numerical artifacts are visible, linked to the numerical diffusion. The results can be used so as to form a tomography gradient for updating the model parameters. When considering Fresnel zones as applied in so-called fat-ray tomography or in finite-frequency tomography, the gradient is generally smoothed at larger scales than the discretization scale, so that the artifacts might not be a complicated issue in practice.

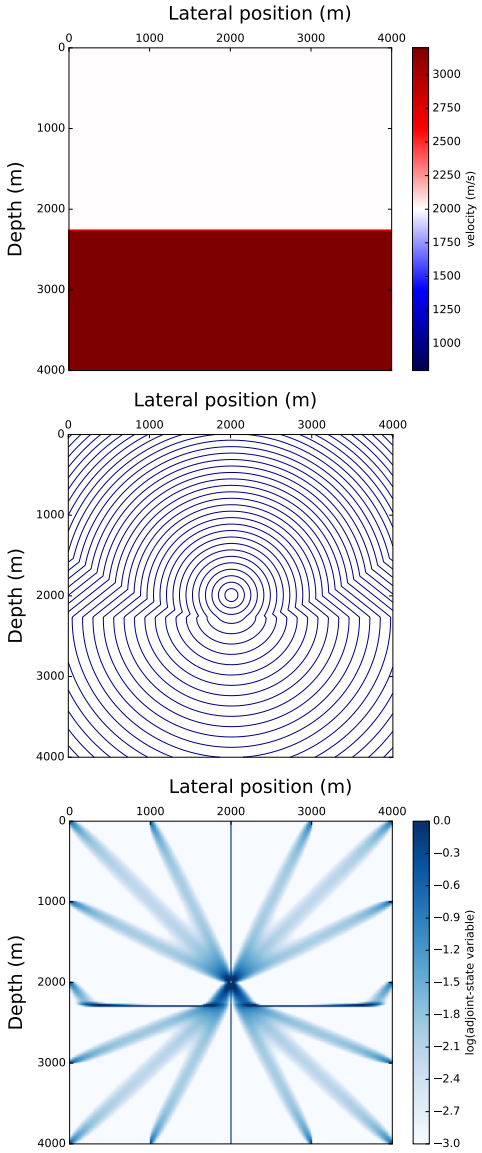


Figure 4.15: Layered model. From top to bottom: Velocity model; isocontours of traveltime; logarithmic adjoint-state variable.

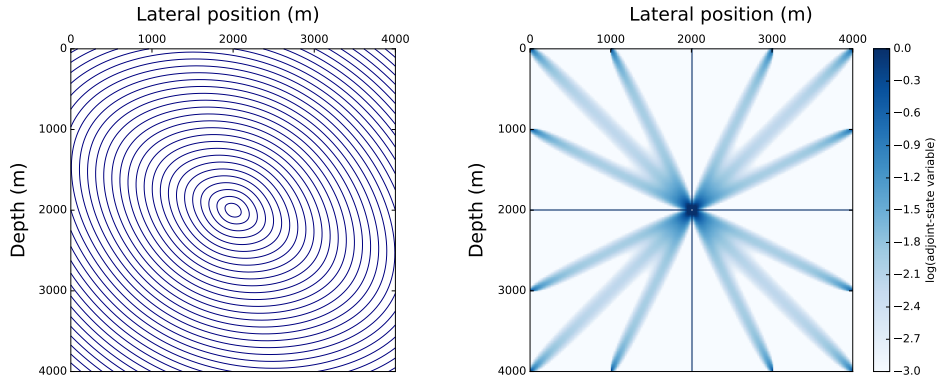


Figure 4.16: Homogeneous TTI model. Left: Isocontours of traveltime. Right: logarithmic adjoint-state variable.

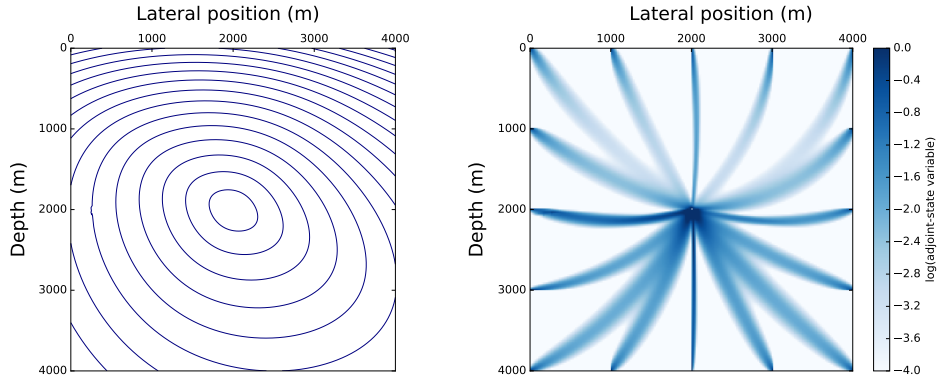


Figure 4.17: TTI model with gradient of vertical velocity. Left: Isocontours of traveltime. Right: logarithmic adjoint-state variable.

4.3 A complex example with topography

In this section I exhibit a final example including the computation of take-off angle, amplitude, and adjoint-state variable using the FSM-DG and FSM-FV solvers in a complex isotropic velocity model with a complex topography. The model is a 2D cross-section extracted from the SEAM II Foothills benchmark model (Regone et al., 2017). The P-wave velocity model is shown in the top plot of figure 4.19. It exhibits complex velocity structures with layering and faulting, as well as near-surface low-velocity regions which are known to impair imaging and inversion results if not considered with care.

The section is 10 km wide and 4 km deep. Inside this model, I consider a point-source located at the surface at $x = 2.700$ km, $z = 0.358$ km. I use a vertically deformed mesh coming from a Cartesian grid, with P^1 (respectively P^0) rectangular elements for traveltime and angle (respectively amplitude and adjoint-state variable) computation and with $N_x = 1001$, $N_z = 400$. For the adjoint-state computation, seismic receivers are modeled all around the domain, at positions listed in table 4.2. Right-hand sides equal to 1 are introduced at each receiver position.

A first computation is performed in a homogeneous model of velocity $c = 2000$ m/s in order to validate the computational behavior in presence of a topography. Computed traveltime, amplitude, angle, and adjoint-state variable fields are shown in figure 4.18. In presence of a topography, the results obtained in this homogeneous medium are consistent with the expectations in terms of traveltime, but also take-off angle, amplitude and adjoint-state variable. The mesh deformation induced by the topography does not seem to impair the results. In particular, it validates the usefulness of the FV approach even when the mesh is not regular: the error introduced by the mesh deformation stays small.

A second computation is then performed in the SEAMII complex velocity model. Computed traveltime, amplitude, angle, and adjoint-state variable fields are shown in figure 4.19. The complexity of the first-arrival traveltime field related to the different branches of the solution results in complex angle, amplitude, and adjoint-state variable solutions.

Receiver positions		
#	x(m)	z(m)
1	0	894
2	5000	489
3	7500	345
4	10000	604
5	0	2000
6	0	3000
7	0	2000
8	10000	3000
9	10000	4000
10	10000	4000
11	0	4000
12	2000	4000
13	4000	4000
14	6000	4000
15	8000	4000
16	10000	4000

Table 4.2: Receiver positions for the adjoint-state variable illustrations.

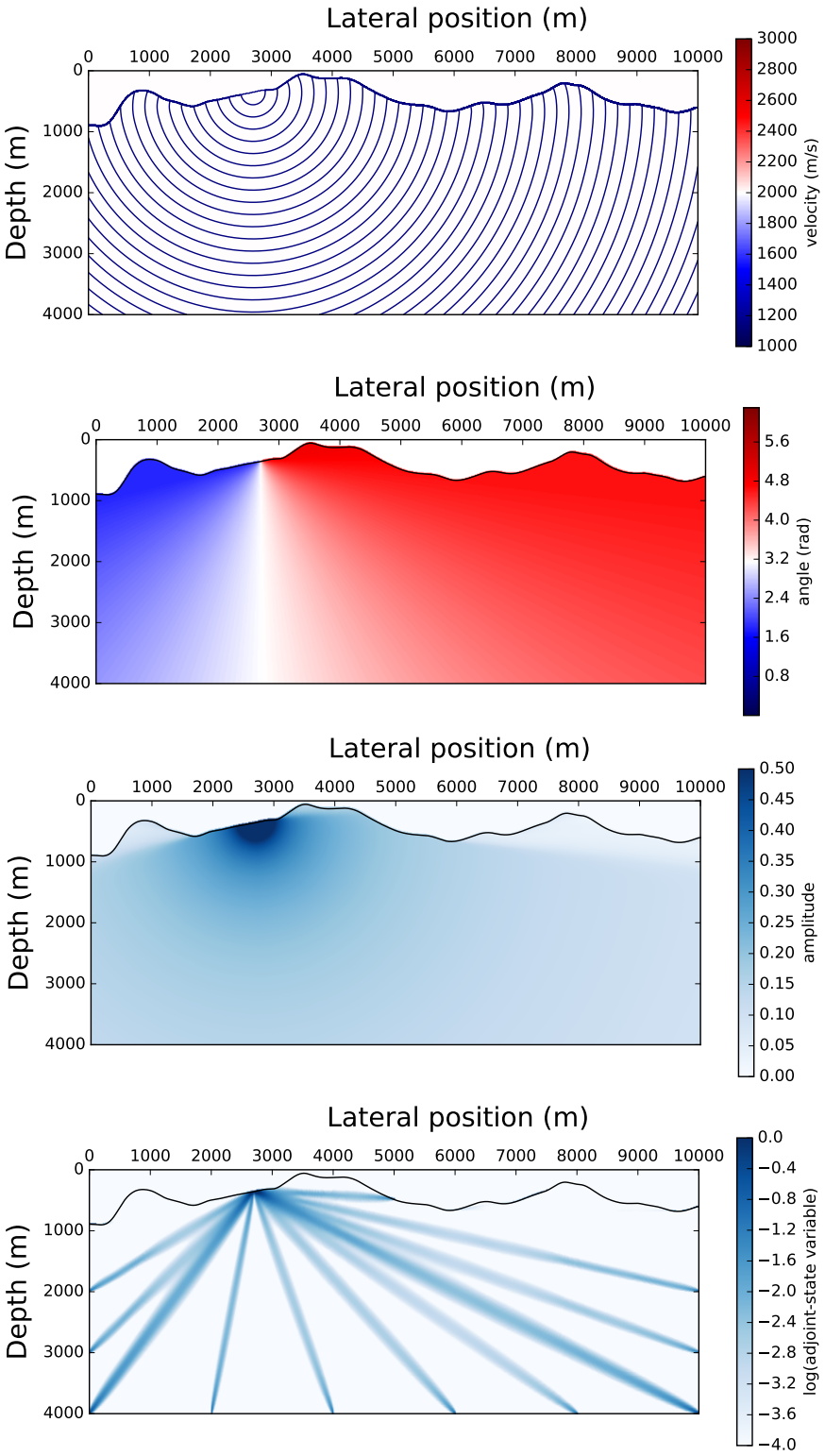


Figure 4.18: Homogeneous model with velocity $c = 2000$ m/s with a topography from a cross-section of the SEAMII Foothills benchmark model. From top to bottom: Isocontours of traveltime; take-off angle; amplitude; logarithmic adjoint-state variable.

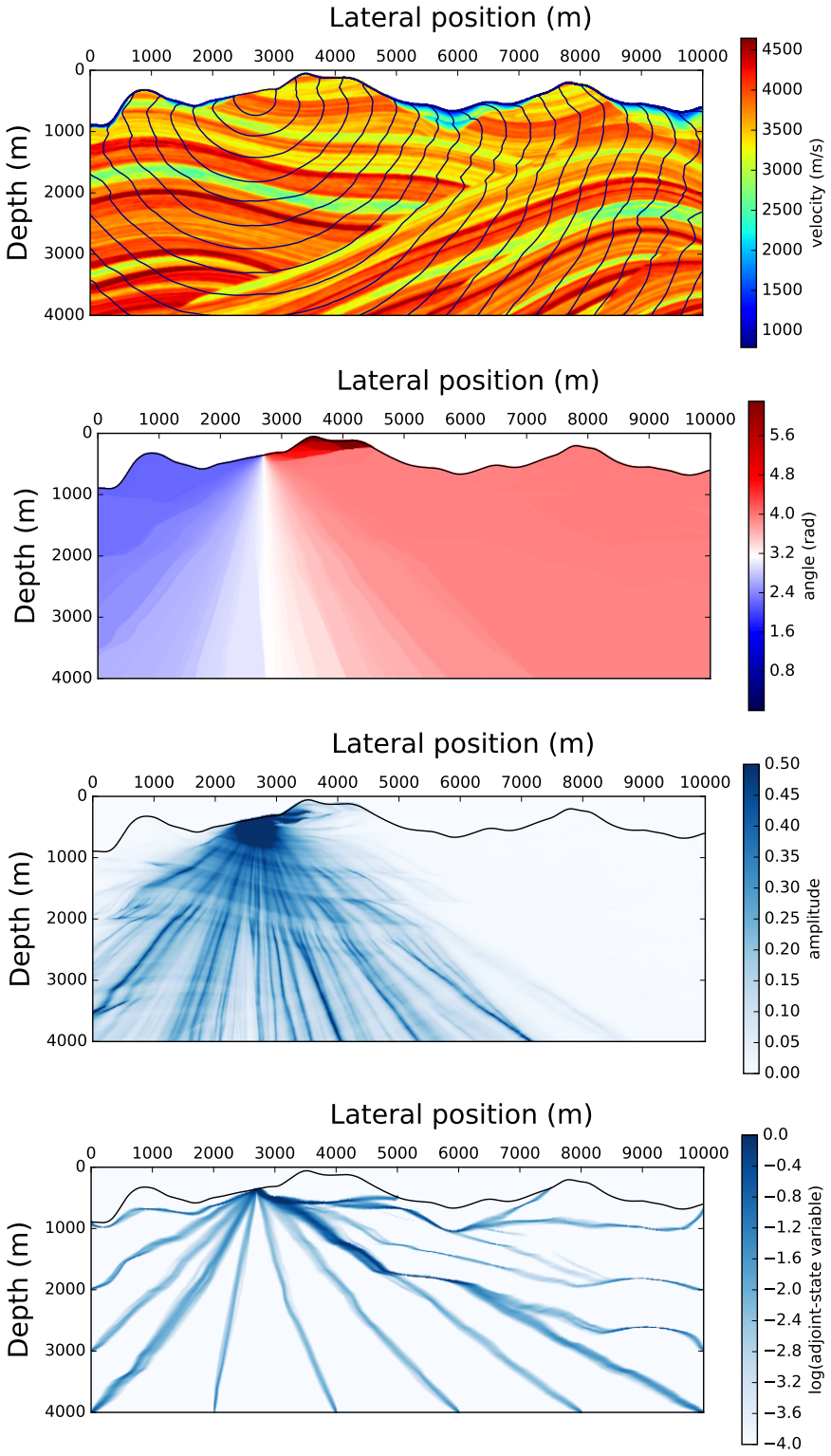


Figure 4.19: Cross-section from SEAMII Foothills benchmark model. From top to bottom: Isocontours of traveltime superimposed over the P-wave velocity model; take-off angle; amplitude; logarithmic adjoint-state variable.

4.4 Conclusion

In this chapter, I have designed numerical algorithms deriving from the FSM-DG Eikonal solver for the computation of other asymptotic quantities in the high-frequency approximation: take-off angle, amplitude, adjoint-state variable. Together with the Eikonal solver, all these solvers form a new consistent, accurate, and efficient set of tools that should be useful in many geophysical applications.

The FSM-DG scheme described in chapter 3 is appropriate for solving the take-off angle equation, since it is a Hamilton–Jacobi equations. For the amplitude and the adjoint-state variable, I choose to adapt it into an FSM-FV solver, because it yields a faster computation and it avoids spurious oscillations near the singularities of the traveltime derivatives. The loss of accuracy compared with a higher-order solver is balanced by the requirements that are generally low in terms of accuracy for these quantities. The amplitude variation is generally accounted for as an order of magnitude with logarithmic units. The adjoint-state variable is employed to build a gradient which is spatially smoothed before applying a model update, depending on the expected resolution related to the first Fresnel zone.

Indeed, these conclusions should be confirmed in practice, by comparing several numerical algorithms performing the same application. Regarding the adjoint-state variable, an adjoint-state tomographic algorithm for anisotropic media will be tested in the near future. Regarding the amplitude and the take-off angle, I have tried to use the solvers in an imaging process. I present the methodologies and some preliminary results in the next chapter.

Chapter 5

Application of the asymptotic solvers to seismic imaging

Contents

Introduction

The main purpose of this chapter is to present the application of the FSM-DG/FSM-FV solvers described in the previous chapters in a geophysical seismic imaging context.

The outline of this chapter is the following:

- In section 5.1, I present the main theoretical ingredients needed for the understanding of the applications I consider next.

- In section 5.2, I consider a first application in the context of migration velocity analysis using an extended domain formulation. This velocity model building method probes the consistency of the background velocity model through the analysis of migrated images. Building these images is computationally intensive: acceleration through efficient techniques is mandatory. Such a technique has been proposed by Chauris and Cocher (2017) using a wave-equation-based formulation. I derive, test and compare an alternative asymptotic formulation. Apart from the validation of the FSM-DG/FSM-FV approach, the results also give an insight on this imaging technique.
- In section 5.3, I consider another configuration, the common-shot formulation. Based on the recent work of Li and Chauris (2018), I combine a wave-equation-based iterative migration scheme with an asymptotic preconditioner. Interestingly, this asymptotic preconditioner has a much simpler implementation than the wave-equation version. I show the resulting benefits of this preconditioning strategy in terms of convergence through two numerical examples.

The work presented in this chapter has been carried out in strong collaboration with the Geophysics team of MINES ParisTech at Fontainebleau, France, and in particular with Hervé Chauris, to whom I am particularly grateful for this work in common.

5.1 Seismic imaging: the main ingredients

In this chapter, I consider the context of seismic linearized modeling and inversion. The linearization of the seismic modeling is the essence of seismic migration, and it consists of a scale separation of the velocity model. A background model, also called reference model or velocity macro-model, contains the long-wavelength components of the velocity model, while the short-wavelength components might be described in two ways:

- a velocity or squared slowness perturbation;
- a distribution of reflectivity.

The linearization of the seismic forward problem is performed by the Born approximation in the former, while the latter employs the Kirchhoff approximation. We consider the first approach in this chapter.

What we call seismic imaging usually refers to two steps. The first one is the estimation of the background model, and the second one is the estimation of the short-wavelength part of the model (velocity perturbation or reflectivity) by depth migration. In a simple case, the background model might be obtained from tomography, for instance, and the migration can be performed once, to obtain an image of the subsurface structure. Alternatively, the two steps can be performed as a repeated sequence, the first one being, for instance, a (migration) velocity analysis, and the second one a standard or iterative migration. Every time the macro-model is estimated, a new velocity perturbation should be derived. Hereafter, only the second step is considered, namely iterative migration.

In this section, I give a quick overview of the Born approximation, as well as the Born inversion, and I introduce the migration/inversion velocity analysis context. More details on general principles of seismic imaging and derivation of Born approximation and inversion from a seismic inverse problem point of view might be found in Lailly (1983) and Tarantola (1984). Beylkin (1985) and Miller et al. (1987) developed an alternative mathematical approach based on an approximate inversion of the high-frequency modeling operator. Bleistein (1987) popularized this direct approach in the geophysical community. A reconciliation of both approaches was first proposed in Jin et al. (1992). Among others, we refer to the review by Symes (2009) for more details.

5.1.1 The first-order Born approximation

Two-scale separation and linearization Consider a constant-density acoustic medium with velocity $c(\boldsymbol{x})$, and a source wavelet $\Omega(\omega)$ in the frequency domain. We define the model parameter $m(\boldsymbol{x})$ to be equal to the squared slowness $1/c^2(\boldsymbol{x})$. The Helmholtz equation holds for the pressure field $p(\boldsymbol{s}, \boldsymbol{x}, \omega)$, yielding

$$\left(m(\boldsymbol{x})(\mathrm{i}\omega)^2 - \Delta\right) p(\boldsymbol{s}, \boldsymbol{x}, \omega) = \Omega(\omega)\delta(\boldsymbol{x} - \boldsymbol{s}), \tag{5.1}$$

for any source point $\boldsymbol{s}$. Here, Δ holds for the Laplacian operator, and δ is the Dirac delta function. We formally define two scales for the representation of long and short wavelengths of the model $m(\boldsymbol{x})$, writing

$$m(\boldsymbol{x}) = m_0(\boldsymbol{x}) + \delta m(\boldsymbol{x}), \tag{5.2}$$

where $m_0(\boldsymbol{x})$ is the macro-model and $\delta m(\boldsymbol{x})$ is the model perturbation. We define $p_0(\boldsymbol{s}, \boldsymbol{x}, \omega)$ as the pressure field obtained in the reference model, thus verifying

$$\left(m_0(\boldsymbol{x})(\mathrm{i}\omega)^2 - \Delta\right) p_0(\boldsymbol{s}, \boldsymbol{x}, \omega) = \Omega(\omega)\delta(\boldsymbol{x} - \boldsymbol{s}). \tag{5.3}$$

We define $\delta p(\boldsymbol{s}, \boldsymbol{x}, \omega)$ such that

$$p(\boldsymbol{s}, \boldsymbol{x}, \omega) = p_0(\boldsymbol{s}, \boldsymbol{x}, \omega) + \delta p(\boldsymbol{s}, \boldsymbol{x}, \omega). \tag{5.4}$$

The first-order Born approximation states that the perturbation of the pressure field $\delta p(\boldsymbol{s}, \boldsymbol{x}, \omega)$ is linearly connected to the model perturbation, when the parameterization is done with the squared slowness. Replacing equations (5.2) and (5.4) inside the Helmholtz equation (5.1) yields

$$\left((m_0 + \delta m)\,(\mathrm{i}\omega)^2 - \Delta\right)(p_0 + \delta p) = \Omega\,(\omega)\,\delta(\boldsymbol{x} - \boldsymbol{s}), \tag{5.5}$$

thus

$$\left(m_0(\mathrm{i}\omega)^2 - \Delta\right) p_0 + \left(m_0(\mathrm{i}\omega)^2 - \Delta\right) \delta p + \delta m\,(\mathrm{i}\omega)^2\, p_0 + (\mathrm{i}\omega)^2\, \delta m\, \delta p = \Omega\,(\omega)\,\delta(\boldsymbol{x} - \boldsymbol{s}). \tag{5.6}$$

Neglecting the second-order term in $\delta m \delta p$ and using equation (5.3), we obtain

$$\left(m_0(\boldsymbol{x})(\mathrm{i}\omega)^2 - \Delta\right) \delta p(\boldsymbol{s}, \boldsymbol{x}, \omega) = -\delta m(\boldsymbol{x})\,(\mathrm{i}\omega)^2\, p_0(\boldsymbol{s}, \boldsymbol{x}, \omega). \tag{5.7}$$

We might rewrite equations (5.3) and (5.7) in the time domain, yielding

$$\begin{cases} \left(m_0(\boldsymbol{x})\dfrac{\partial^2}{\partial t^2} - \Delta\right) p_0(\boldsymbol{s}, \boldsymbol{x}, t) = \Omega\,(t)\,\delta(\boldsymbol{x} - \boldsymbol{s}), \\ \left(m_0(\boldsymbol{x})\dfrac{\partial^2}{\partial t^2} - \Delta\right) \delta p(\boldsymbol{s}, \boldsymbol{x}, t) = -\delta m(\boldsymbol{x})\dfrac{\partial^2 p_0}{\partial t^2}(\boldsymbol{s}, \boldsymbol{x}, t). \end{cases} \tag{5.8}$$

The perturbation δp verifies the same scalar wave equation as p_0 with a different source term: δp is excited by a source at the points of the space where $\delta m \neq 0$ and at times when the second-order derivative of the pressure field is non-zero. Therefore, any numerical algorithm suitable for the modeling of p_0 in the background model will also be suitable for the modeling of δp under the Born approximation, provided p_0 is known.

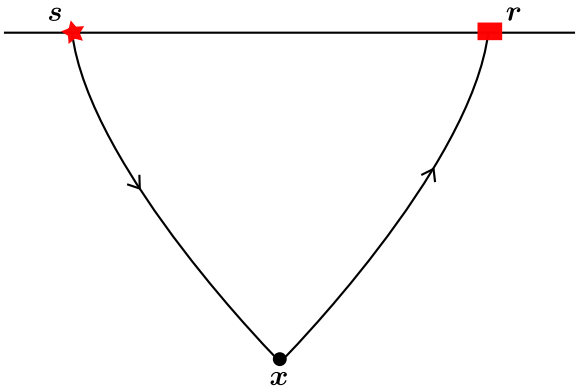


Figure 5.1: Single contribution of one point $\boldsymbol{x}$ at receiver $\boldsymbol{r}$ for a source $\boldsymbol{s}$, under the Born single-scattering approximation.

Introducing the Green's functions The Green's function G_0 is the impulse response of the background medium, defined by

$$\left(m_0 \frac{\partial^2}{\partial t^2} - \Delta\right) G_0(\boldsymbol{s}, \boldsymbol{x}, t) = \delta\left(t\right) \delta(\boldsymbol{x} - \boldsymbol{s}), \tag{5.9}$$

or, in the frequency domain,

$$\left(m_0 (\mathrm{i}\omega)^2 - \Delta\right) G_0(\boldsymbol{s}, \boldsymbol{x}, \omega) = \delta(\boldsymbol{x} - \boldsymbol{s}). \tag{5.10}$$

Using equation (5.10), an explicit expression of the Born data at receiver locations $\boldsymbol{r}$ with respect to the model perturbation can be derived:

$$\delta p(\boldsymbol{s}, \boldsymbol{r}, \omega) = -\left(\mathrm{i}\omega\right)^2 \Omega(\omega) \int G_0(\boldsymbol{s}, \boldsymbol{x}, \omega)\, G_0(\boldsymbol{x}, \boldsymbol{r}, \omega)\, \delta m(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}. \tag{5.11}$$

The single contribution of one point $\boldsymbol{x}$ at receiver $\boldsymbol{r}$ for a source $\boldsymbol{s}$ is sketched in figure 5.1. The scattered wavefield is excited by the forward source wavefield. The multiple scattering effects are neglected under the Born approximation.

5.1.2 Born inversion

Here, I give only the main results that are commonly used in seismic imaging. Theoretical derivations might be found in Lailly (1983) and in Tarantola (1984). We consider an observed reflected dataset d_{obs}, a calculated dataset d_{cal}, and the ℓ^2 cost function written as

$$\mathcal{C}[\delta m] = \frac{1}{2} \| d_{\mathrm{obs}}(\boldsymbol{s}, \boldsymbol{r}, \omega) - d_{\mathrm{cal}}[\delta m](\boldsymbol{s}, \boldsymbol{r}, \omega) \|^2. \tag{5.12}$$

The calculated data d_{cal} are linearized data under the Born approximation. Note that the same approach applies for the non-linear case, leading to the so-called full-waveform inversion (Virieux and Operto, 2009; Virieux et al., 2017).

The solution of the linearized inverse problem is the model perturbation that minimizes the cost function (5.12). It involves the gradient and the inverse Hessian of the cost function, that are first- and

second-order derivatives of the cost function with respect to the model parameters. Basic migration algorithms do not account for the Hessian but only for the gradient which writes

$$\frac{\partial \mathcal{C}}{\partial \delta m}(\boldsymbol{x}) = -\iiint \mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{r}\mathrm{d}\omega(\mathrm{i}\omega)^2\Omega^*(\omega)G_0^*(\boldsymbol{s},\boldsymbol{x},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)G_0^*(\boldsymbol{x},\boldsymbol{r},\omega), \tag{5.13}$$

where $\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)$ denotes the misfit $d_{\text{obs}} - d_{\text{cal}}$. In this well-known expression, we can recognize the frequency-domain formula of zero-lag cross-correlation of the incident wavefield and a back-propagated residual wavefield from the receivers[1].

Utilizing only the gradient in the inversion does not provide the solution of the minimization problem for (5.12). Yet, it provides a first structural image of the subsurface. For a quantitative image of the model perturbation, one would need to iterate using local optimization solver. Quasi-Newton methods (Lambaré et al., 1992) and conjugate-gradient methods have been proposed (Pica et al., 1990; Crase et al., 1990) for instance. Accelerating these solvers through preconditioning might require the access to an approximation of the inverse Hessian operator. Diagonal approximations have been proposed (Tarantola, 1984; Plessix, 1996). High-frequency asymptotic approximations of the inverse Hessian have also been considered, which are at the basis of ray-based quantitative migration methods (Beylkin, 1985; Bleistein, 1987; Beylkin and Burridge, 1990; Jin et al., 1992; Lambaré et al., 2003; Métivier et al., 2015, the last one combines an asymptotic migration with a full-waveform inversion). Finally, pseudo-inverse approaches have been recently derived in the context of migration velocity analysis, that I present next. They are pseudo-inverse in the sense that the inverses are derived in the asymptotic approximation. However, the final formulas do not contain asymptotic quantities: only wave-equation operators such as time or spatial gradients are used.

5.1.3 Migration velocity analysis

The computation of Green's functions required in the migration formula (5.13) implies that the background model is known. This is not the case in practice, and the discrepancies between the background model and the true medium affects images. More precisely, the quality of the background model can be evaluated by measuring the focusing, or consistency, of migrated images, using various techniques referred to as migration velocity analysis (MVA) techniques. Traditionally, the data is split into subsets, such as common-shot or common-offset gathers. These gathers are migrated separately in the same background model, and the resulting images should be consistent if the background model is accurate enough (see Symes, 2008, for a review).

Instead of splitting the data, another approach introduces a spatial or temporal shift in the migration formula, yielding an extension of equation (5.13), and the full data set is migrated at once (Faye and Jeannot, 1986; de Bruin et al., 1990; Biondi and Symes, 2004; Sava and Fomel, 2006; Symes, 2008). One might consider a spatial horizontal shift, known as subsurface offset or space lag, and denoted by $\boldsymbol{h}$ in the following. One dimension is added to the image domain, therefore called the extended domain. Thus, the data space and the model space have the same dimension (see Chauris and Cocher, 2017, for instance). It becomes then possible, in theory, to reconstruct the data from the reflectivity/model perturbation, even if the background model is inaccurate (Symes, 2008). The Born modeling operator (5.11) and the migration (gradient) operator (5.13) now write

$$\delta p(\boldsymbol{s},\boldsymbol{r},\omega) = -(\mathrm{i}\omega)^2\,\Omega(\omega)\iint G_0(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)\,G_0(\boldsymbol{x}+\boldsymbol{h},\boldsymbol{r},\omega)\,\delta m(\boldsymbol{x},\boldsymbol{h})\,\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{h}, \tag{5.14}$$

[1] This is very close to the imaging principle given by Claerbout (1971).

$$\frac{\partial \mathcal{C}}{\partial \delta m}(\boldsymbol{x},\boldsymbol{h}) = -\iiint \mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{r}\mathrm{d}\omega(\mathrm{i}\omega)^2\Omega^*(\omega)G_0^*(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)G_0^*(\boldsymbol{x}+\boldsymbol{h},\boldsymbol{r},\omega). \quad (5.15)$$

Once data have been migrated, a second cost function indicates how focused the energy is in the extended domain. This ℓ^2 cost function may write

$$\mathcal{C}_{\mathrm{VA}}[m_0] = \frac{1}{2}\| \, |\boldsymbol{h}| \, \delta m\|^2, \quad (5.16)$$

where δm has been obtained during the migration step. The velocity analysis step consists of minimizing this cost function, in order to concentrate the energy around $\boldsymbol{h} = 0$, thus improving the quality of the background model.

For a stable velocity analysis, recent recommendation is to perform the analysis on quantitative migrated images, i.e. by properly minimizing the cost function defined in equation (5.12). This is referred to as inversion velocity analysis (IVA), and it has been proposed recently in the subsurface-offset formulation (Chauris and Cocher, 2017; Hou and Symes, 2017), as well as in the common-shot configuration (Li and Chauris, 2018).

In practical applications, each iteration of the iterative migration is computationally expensive, and the process might converge slowly. In this context, pseudo-inverses for the Born modeling operator have been proposed as an alternative to iterative migration (ten Kroode, 2012; Hou and Symes, 2015, 2017; Chauris and Cocher, 2017). The main effects of the approximate Hessian are the source wavelet deconvolution and the compensation for the geometrical amplitude decay, as well as for the uneven illumination of the subsurface. Such inverses are excellent preconditioners for iterative migration (Hou and Symes, 2016), and lead to more stable MVA schemes (Chauris and Cocher, 2017; Hou and Symes, 2018). A comparison between the different formulations is discussed in Chauris and Cocher (2018). They are all equivalent in the asymptotic sense. All of them have been implemented in a wave-equation approach.

In the two next sections, I present two different configurations for these approximate direct inversions. In section 5.2, I consider the inverse formulation proposed in Chauris and Cocher (2017), in the subsurface-offset extended domain, and I derive, test and compare the alternative asymptotic formulas. The first goal of this section, following on from the previous chapters, is the validation of the potential use of FSM-DG/FSM-FV solvers in an imaging application. Besides, it is also an opportunity to validate the asymptotic formulation itself, which in turn gives insight on the behavior and the applicability of the original wave-equation-based formula from Chauris and Cocher (2017).

In section 5.3, I go back to the common-shot formulation, which has the strong advantage of requiring less memory than the subsurface-offset extended domain. This could be a main concern when considering 3D extension. Moreover, this formulation is shot-oriented as the acquisition itself: two nearby shots already provide some information on the macro-model, while for the subsurface-offset configuration, all shots should be migrated first. I show how to build a numerical strategy based on this common-shot configuration, combining the recent pseudo-inverse approach from Li and Chauris (2018) with asymptotic approximations. The first goal of this section is to give another illustration of the interest of asymptotic quantities in imaging. Moreover, the preliminary results presented in this section are quite interesting, and should encourage us to conduct further investigations on this asymptotic preconditioning strategy in the future.

5.2 Born inversion in the subsurface-offset extended domain: a comparison of approaches

5.2.1 An approximate Born inverse operator

A recent expression for the inverse of the Born modeling operator has been proposed in Chauris and Cocher (2017). This is the one I present here. Consider a velocity perturbation $\xi(\boldsymbol{x},\boldsymbol{h}) = \delta v(\boldsymbol{x},\boldsymbol{h})$ in the extended domain. Following Chauris and Cocher (2017), we write

$$\begin{aligned}
\bar{F}(\xi)(\boldsymbol{s},\boldsymbol{r},\omega) &= (\mathrm{i}\omega)^2\Omega(\omega)\iint \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{h}\, G_0(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)\times \\
&\quad \frac{2}{v_0^3(\boldsymbol{x})}\xi(\boldsymbol{x},\boldsymbol{h})G_0(\boldsymbol{x}+\boldsymbol{h},\boldsymbol{r},\omega), \qquad (5.17)\\
\bar{F}^T(\Delta d)(\boldsymbol{x},\boldsymbol{h}) &= \frac{2}{v_0^3(\boldsymbol{x})}\iiint \mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{r}\mathrm{d}\omega\,(\mathrm{i}\omega)^2\Omega^*(\omega)\times \\
&\quad G_0^*(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)G_0^*(\boldsymbol{x}+\boldsymbol{h},\boldsymbol{r},\omega), \qquad (5.18)\\
\bar{F}^\dagger(\Delta d)(\boldsymbol{x},\boldsymbol{h}) &= -16v_0(\boldsymbol{x})\partial_z\iiint \mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{r}\mathrm{d}\omega\,\frac{1}{\mathrm{i}\omega}\Omega^\dagger(\omega)\times \\
&\quad \partial_{s_z}G_0^*(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)\partial_{r_z}G_0^*(\boldsymbol{x}+\boldsymbol{h},\boldsymbol{r},\omega), \qquad (5.19)
\end{aligned}$$

where $\bar{F}$, $\bar{F}^T$ and $\bar{F}^\dagger$ stand for the extended Born modeling, adjoint (standard migration) and inverse operators, respectively. The data d depends on the source, receiver and angular frequency $(\boldsymbol{s},\boldsymbol{r},\omega)$. The spatial coordinates and subsurface offset are denoted by $(\boldsymbol{x},\boldsymbol{h})$. G_0 is the Green's function computed in the reference background model v_0, while Ω and $\Omega^\dagger = \Omega^*/||\Omega||^2$ are the seismic wavelet and its inverse, in the frequency domain.

The main differences between the expressions for the adjoint, equation (5.18), and inverse approaches, equation (5.19), lie in the time and spatial derivatives. The derivatives ∂_{s_z} and ∂_{r_z} acting on the Green's functions are equivalent in an asymptotic sense to a multiplication by $\mathrm{i}\omega\cos\beta_{\boldsymbol{s}}/v_0(\boldsymbol{s})$ and $\mathrm{i}\omega\cos\beta_{\boldsymbol{r}}/v_0(\boldsymbol{r})$, where $(\beta_{\boldsymbol{s}},\beta_{\boldsymbol{r}})$ are the emerging angles at the surface (see figure 5.2). This comes from considering the following asymptotic Green's function approximation:

$$G_0(\boldsymbol{s},\boldsymbol{x},\omega) \simeq A_0(\boldsymbol{s},\boldsymbol{x})\,k(\omega)\mathrm{e}^{\mathrm{i}\omega\,T_0(\boldsymbol{s},\boldsymbol{x})}, \qquad (5.20)$$

where the $k(\omega)$ integration term depends on the dimension. In 2D we have $k(\omega) = 1/\sqrt{\mathrm{i}\omega}$. T_0 and A_0 denote the asymptotic traveltime and amplitude at point $\boldsymbol{x}$ for a point source located at $\boldsymbol{s}$. The spatial derivation of (5.20) is approximated by terms in $\mathrm{i}\omega$ because the amplitude term has a smoother behavior than the oscillating term $\mathrm{e}^{\mathrm{i}\omega\,T_0(\boldsymbol{s},\boldsymbol{x})}$, which is the essence of high-frequency approximations.

The same analysis can be conducted for the derivative ∂_z acting at the image point. As a consequence, the pseudo-inverse cannot reconstruct data with grazing waves nor diving waves for which the cosine values would be zero.

The main steps for the derivation of the inverse are the followings (ten Kroode, 2012; Hou and Symes, 2015, 2017; Chauris and Cocher, 2017). First replace the Green's function by their asymptotic expression and apply the stationary phase approximation to evaluate the specular contributions. The determinant between the acquisition coordinates and the extended wave number domain plays an essential role to remove the geometrical amplitude factors on the source and receiver sides (Zhang et al.,

2005b; ten Kroode, 2012; Hou and Symes, 2016). The partial derivation ∂_z in front of the integral in equation (5.19) is a simplified version valid when the macro-model is locally invariant (ten Kroode, 2012; Hou and Symes, 2015, 2017; Chauris and Cocher, 2017). The expressions are derived for the acoustic constant density case.

In the next section, I present two implementations of the approximate Born inverse operator (5.19). The first one is the classical wave-equation approach, and the second one is an asymptotic approach which makes use of the asymptotic solvers developed in the previous chapters. I will refer to the latter as the Eikonal implementation.

5.2.2 Implementation

5.2.2.1 Wave-equation approach

We use here a standard staggered-grid finite-difference approach, with 10 points per wavelength for the spatial discretization (Virieux, 1986b). For efficiency reasons and as for the standard migration case, equation (5.19) is implemented in a source oriented manner and rewritten as a cross-correlation between the forward wave field $S(\boldsymbol{s},\boldsymbol{x},\omega)$ and the residual wave field $R(\boldsymbol{s},\boldsymbol{x},\omega)$. Let $\mathcal{L}_0$ be the wave equation operator in the velocity model v_0. Then S and R are solutions of

$$\mathcal{L}_0 S = \frac{\Omega^\dagger(\omega)}{i\omega}\frac{\partial\delta(\boldsymbol{x}-\boldsymbol{s})}{\partial z}, \tag{5.21}$$

$$\mathcal{L}_0^* R = \int \mathrm{d}\boldsymbol{r}\frac{\partial\delta(\boldsymbol{x}-\boldsymbol{r})}{\partial z}\Delta d(\boldsymbol{s},\boldsymbol{r},\omega). \tag{5.22}$$

The $*$ indicates a back-propagation. It means in practice that S and R are solution of the wave equation for dipole sources and dipole data residuals. Then, S and R are cross-correlated for different horizontal subsurface offsets $\boldsymbol{h}=(h,0)$:

$$\bar{F}^\dagger(\Delta d)(\boldsymbol{x},\boldsymbol{h}) = -16 v_0(\boldsymbol{x})\partial_z \iint \mathrm{d}\boldsymbol{s}\mathrm{d}\omega\, S^*(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)R(\boldsymbol{s},\boldsymbol{x}+\boldsymbol{h},\omega). \tag{5.23}$$

5.2.2.2 Eikonal approach

In the Eikonal implementation, we make use of the ray ansatz to replace the Green's functions by their asymptotic approximations under the form of expression (5.20). In the presence of multiple phases (e.g. wavefield triplication), the asymptotic Green's functions should be computed as the sum of contributions in amplitude and traveltime from each phase. Here, we may compute T_0 and A_0 using the Eikonal and transport solvers presented in the previous chapters, and considering only first arrivals.

The Eikonal implementation requires the computation of traveltime maps from all the sources and receivers, as well as the corresponding amplitude maps, in order to reconstruct asymptotic Green's functions (5.20). Traveltime maps are computed using the FSM-DG solver for the Eikonal equation (see chapter 3), and amplitude maps are computed using the FSM-FV solver for the transport equation (see section 4.2.1).

In the approximate Born inverse operator (5.19), the spatial derivation of the Green's functions is replaced by a multiplication by $\mathrm{i}\omega\boldsymbol{p}$ in the asymptotic formulation. This comes directly from the asymptotic expression (5.20). Thus, the vertical derivative at the image point implies the vertical component

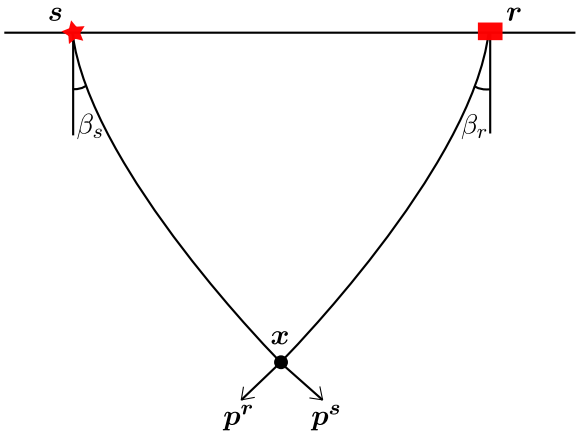


Figure 5.2: Illustration of the quantities involved in the asymptotic inverse formula (5.24). Angles $\beta_{\boldsymbol{s}}$ and $\beta_{\boldsymbol{r}}$ are the take-off angles at source $\boldsymbol{s}$ and receiver $\boldsymbol{r}$. Quantities $p_z^{\boldsymbol{s}}$ and $p_z^{\boldsymbol{r}}$ are the vertical components of the slowness vectors $\boldsymbol{p}^{\boldsymbol{s}}$ and $\boldsymbol{p}^{\boldsymbol{r}}$ at the image point $\boldsymbol{x}$.

p_z of the slowness vector at depth, whereas the vertical derivative with respect to the source/receiver position implies the cosine of the emerging angle at the surface. Therefore, maps of p_z and emerging angles are also pre-computed for each source and receiver position. Instead of taking a posteriori spatial gradient of traveltime maps with FD operators, the p_z component is directly accessed when computing a DG polynomial solution for the traveltime, by extracting the z-derivative of the traveltime field. Angle geometry is shown in figure 5.2. The emerging angle corresponds to the take-off angle as defined in section 1.1.2.3. Therefore, it can be computed using the FSM-DG solver for the Hamilton–Jacobi take-off angle equation (see section 4.1).

After the computation of all the required maps, the inversion step consists of summing the contribution of each trace in the extended image. The Born inverse operator (5.19) now writes

$$\begin{aligned}\bar{F}^{\dagger}(\Delta d)(\boldsymbol{x},\boldsymbol{h}) = \frac{32}{v_0(\boldsymbol{x})}\iiint \mathrm{d}\boldsymbol{s}\mathrm{d}\boldsymbol{r}\mathrm{d}\omega\,(\mathrm{i}\omega)^2\Omega^{\dagger}(\omega)\cos\beta_{\boldsymbol{s}}\cos\beta_{\boldsymbol{r}}(p_z^{\boldsymbol{s}}+p_z^{\boldsymbol{r}})\times \\ G_0^*(\boldsymbol{s},\boldsymbol{x}-\boldsymbol{h},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)G_0^*(\boldsymbol{x}+\boldsymbol{h},\boldsymbol{r},\omega),\end{aligned} \tag{5.24}$$

where $\beta_{\boldsymbol{s}}$ and $\beta_{\boldsymbol{r}}$ stand for the take-off angles computed for sources and receivers at the surface, and $p_z^{\boldsymbol{s}}$ and $p_z^{\boldsymbol{r}}$ denote the vertical components of the slowness at the image point (at depth) when considering the source side and the receiver side, respectively.

Next, I exhibit numerical results obtained with both implementations of the approximate extended Born inverse operator.

5.2.3 Numerical illustration

In this section, I give a numerical illustration of the approximate extended Born inverse operator in order to compare the two implementations: the wave-equation approach (denoted by WEQ), and the Eikonal approach (denoted by EIK). The main purposes are the followings:

- Validate the use of the FSM-DG/FSM-FV solvers in an imaging application;
- Validate the EIK implementation, since this is the first time it is implemented from the formulation of Chauris and Cocher (2017);

- Test the applicability of the inverse in a simple example with a triplicated wavefield;
- Test the aliasing effect due to sparse data.

For these purposes, a reference macro-model is built inside a physical domain of 1800 m width and 1200 m depth, and discretized on a finite-difference grid with 301 points in the horizontal direction and 201 points in the vertical direction. This discretization is performed so that sources and receivers, located at the surface along the entire profile, are distributed every two grid points, thus every 12 m. The receiver positions are symmetrical with respect to the source position, with offsets from -720 to $+720$ m. The reference macro-model contains a low velocity zone, responsible for triplication of the wavefield. The triplication is visible in figure 5.3. Note that rays are only used in this figure as an illustration tool. They are not considered in the WEQ nor EIK approaches. The exact reflectivity model consists of four diffraction points localized at $x = 900$ m and at depths $z = 180, 420, 660$, and 900 m. The source signal is a Ricker signal with a maximum frequency $f_{\text{max}} = 40$ Hz.

The observed data are computed within the WEQ approach, using the extended Born modeling operator (5.17), and plotted in figure 5.4a for a central source position. Compared to the classical Born modeling in the standard space, modeling in the extended domain implies an additional integration over all subsurface offsets.

For the direct inversion, we consider that the exact velocity macro-model is known. We then use the same velocity macro-model (figure 5.3), and two different implementations: the WEQ approach with operator (5.23), and the EIK approach with operator (5.24) using the asymptotic Green's function expression (5.20). The computations of the asymptotic quantities required in the EIK approach are performed in a 201×301 regular mesh. The center of each cell of the mesh coincides with a point of the FD grid. DG traveltimes, angles, and FV amplitudes are extracted at these FD grid locations. The DG mesh size used here has been directly derived from the FD grid size. This size ensures that the DG computations are at least as accurate as FD computations on the FD grid, because the DG method is higher-order. This mesh could probably be decimated with only few effect on the results. However, I did not investigate further on this specific topic. For large-scale 3D applications, in addition to the decimation of sources and receivers with subsequent interpolation, the decimation of the DG mesh would yield lower computational cost and memory requirements for the map computation.

The inversion yields a squared slowness perturbation field, plotted in figure 5.5 for both WEQ and EIK approaches. The two shallower scatterers are retrieved accurately, showing a good focusing of the squared slowness perturbation around the scatterer positions, and in a similar way for both approaches. The two deeper scatterers are affected by the triplication. Since only the first arrival is modeled in the EIK approach, the focusing around the scatterers is worse in this case than in the WEQ approach.

After the inversion, the reconstructed data are obtained thanks to a modeling step with operator (5.17), using either a wave-equation modeling to compute the Green's functions (WEQ approach), or the asymptotic Green's functions formula (5.20) resulting in a Ray+Born formula (EIK approach). The reconstructed data is shown for a central source in figure 5.4b for the WEQ approach, and in figure 5.4c for the EIK approach. All three datasets in figure 5.4 are displayed at the same scale. The effect of the triplication is visible on the observed data (figure 5.4a), and it is well retrieved in the reconstructed data in the WEQ approach. However, in the EIK approach, the energy corresponding to the shallower scatterers is underestimated, due to the fact that it only accounts for the first arrival. Extracted traces for offset zero and offset 360 m show an excellent match (figure 5.6), except for the two deeper scatterers in the EIK approach. The WEQ approach suffers from small numerical dispersion, explaining the little kinematic mismatch, for example around 0.25 s, between the observed data (WEQ solution) and

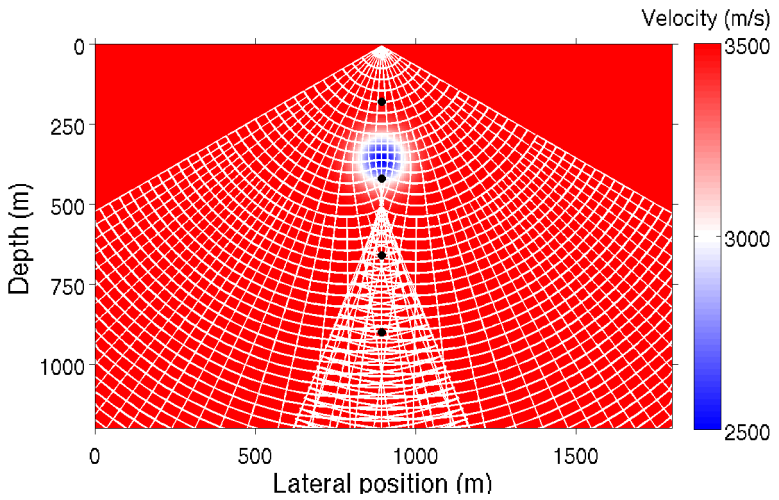
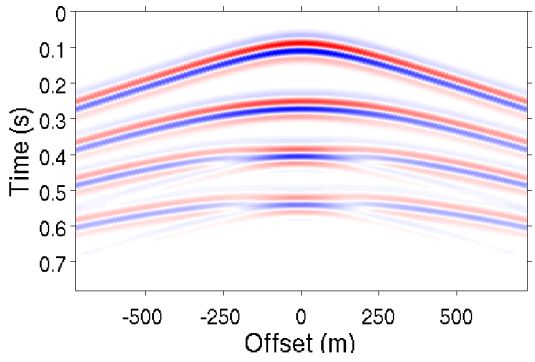


Figure 5.3: Ray paths and isocontours of traveltime (in white lines) for a source at 900 m, superimposed on the background velocity model. The position of the four scatterers are indicated by the black dots. Note that the rays are only used here to illustrate the triplication, but are not considered in the WEQ nor EIK numerical approaches.

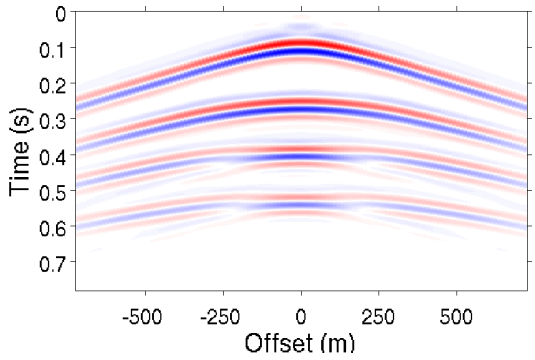
the Eikonal-based reconstructed shot. Near the source, the amplitude term in the Eikonal solution is estimated by $1/\sqrt{r}$, where r is the distance to the source, but this does not lead to discrepancies.

The computation is parallelized per source. In the WEQ approach, the computation took approximately 120 minutes using one cluster node of two eight-core Intel® Xeon® E5-2670 processors with frequency 2.6 GHz. The EIK computation took approximately 210 minutes on a node with the same specifications. Note that these are only orders of magnitude. In particular, the EIK code could be optimized a lot in order to save computational cost. For instance, the traveltime, amplitude, and angle maps are computed in the whole domain, while only a part of it is considered for each source location due to the offset limits.

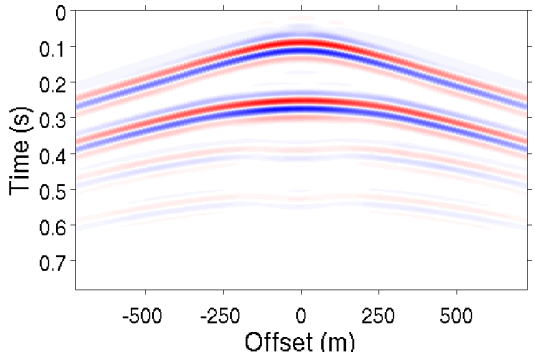
We now evaluate the impact of acquisition sparsity. A decimation factor $\Delta_{\text{acq}} \in [2, 5, 10, 15, 30]$ is defined. A given factor means that we consider sources and receivers every Δ_{acq} points on the grid used for the wave-equation approach. It means that if Δ_{acq} is multiplied by a factor α, there are α^2 less traces (twice less sources and receivers). Case $\Delta_{\text{acq}} = 2$ is the reference case discussed so far. For values of Δ_{acq} up to 10 (figures 5.5 and 5.7a and b), the reflectivity sections are very similar, both for WEQ and EIK approaches. Here, we display the summation over all 61 subsurface offsets as proposed in Hou and Symes (2015). For larger increments ($\Delta_{\text{acq}} \in [15, 30]$), aliasing is visible, especially at shallow depths, as expected. The associated reconstructed data for the wave-equation approach suffers from aliasing for $\Delta_{\text{acq}} \geq 15$ (figure 5.8). Significant differences are observed for $\Delta_{\text{acq}} = 30$ (figure 5.8d). The same conclusion holds for the Eikonal approach (not shown here). The increment can be related to the characteristic wavelength λ_c. For distances between sources and between receivers larger than the typical wavelength (i.e. $\Delta_{\text{acq}} > 10$, here), the direct scheme suffers from aliasing. An anti-alias filter, as proposed in Abma et al. (1999) and in Baina et al. (2003), should be added in expression (5.19). These filters examine the slopes at source and receiver, as well as the slopes in the data, in order to determine the maximum frequency that can be migrated without aliasing. For that purpose, the quantities provided by the Eikonal solution are very useful.



(a)



(b)



(c)

Figure 5.4: a) Observed shot gathers for source position 900 m computed with the wave-equation solution, and reconstructed shots for the WEQ (b) and EIK (c) approaches. All images are displayed at the same scale.

5.2.4 Conclusion

The two implementations, based on the wave equation and on the Eikonal approach, are really different in practice, involving very different intermediate steps (FD wavefield propagation/backpropagation versus FSM-DG/FSM-FV asymptotic computations, for instance). Therefore, the perfect compatibility between both approaches in zones that are not affected by triplications, even if predictable by the theory, is an appreciable result in practice. This, in particular, validates the use of the FSM-DG/FSM-FV solvers in applications.

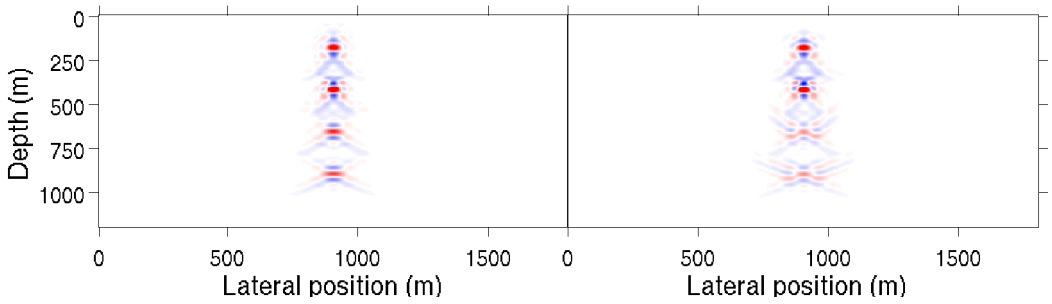


Figure 5.5: Squared slowness perturbation obtained by summing over all subsurface offsets, displayed at the same scale. Left: WEQ approach, right: EIK approach.

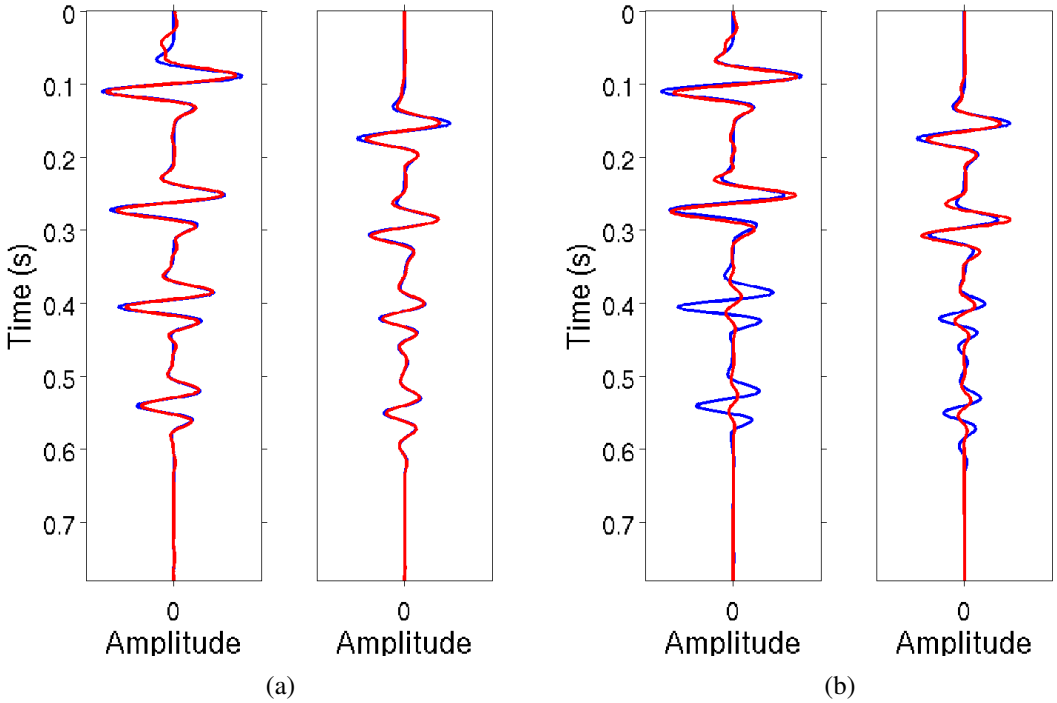


Figure 5.6: Traces extracted from figure 5.4 for offsets 0 and 360 m. a) WEQ approach, b) EIK approach. Blue: observed shot, red: reconstructed shot.

Besides, a second result is the validation of the behavior of the wave-equation based approximate inverse even in presence of triplication. Even if this was considered in the work of ten Kroode (2012), this was not obvious, since the mathematical derivation of this inverse involves the use of asymptotic quantities.

Finally, for sparse data, an anti-alias filter should be incorporated. The Hamilton–Jacobi formalism could be very useful in that context to determine the optimal parameters (Abma et al., 1999; Baina et al., 2003).

In the extended domain formulation, it is possible, in theory, to reconstruct the data from the reflectivity/model perturbation, even if the background model is inaccurate (Symes, 2008). However, a severe drawback is the extra-cost of this approach. On the other hand, the common-shot formulation requires much less memory, and is suitable for 3D extension. In the next section, I show how to build

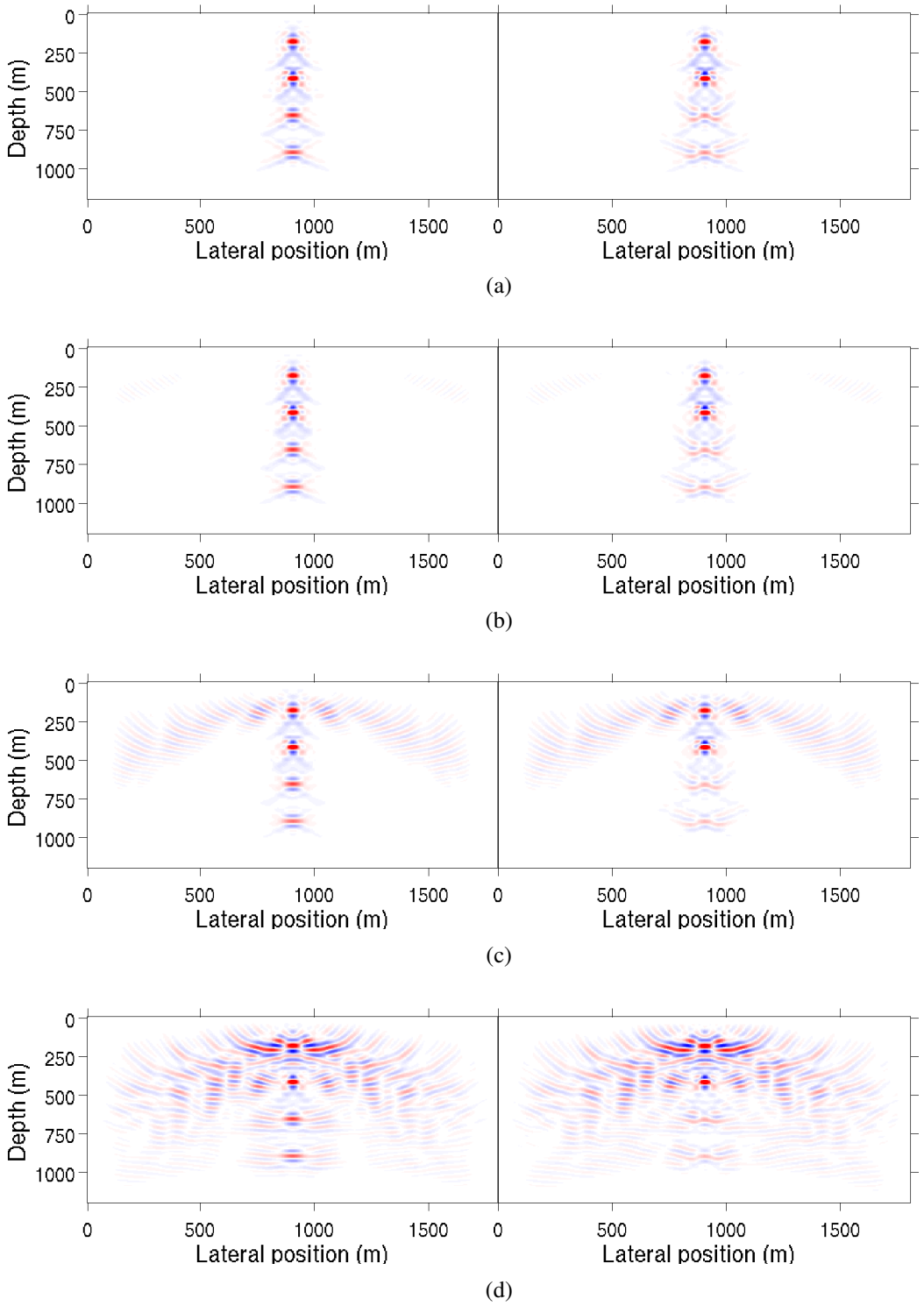


Figure 5.7: Squared slowness perturbation obtained by summing over all subsurface offsets, for increments $\Delta_{\text{acq}} \in [5, 10, 15, 30]$ (a–d), all displayed at the same scale. Left: WEQ approach, right: EIK approach. For $\Delta_{\text{acq}} \in [15, 30]$ (c and d), the images suffer from the sparse acquisition sampling.

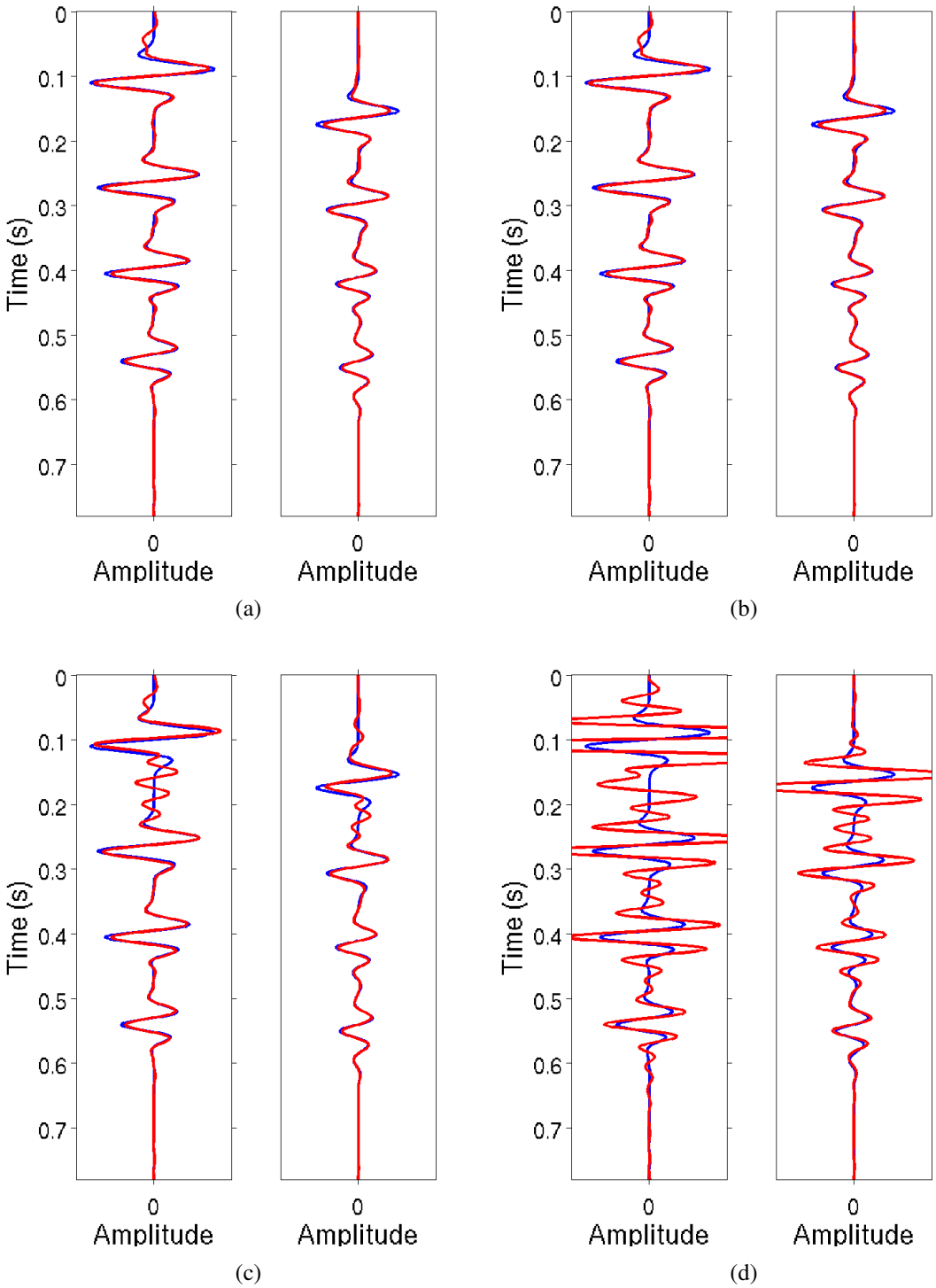


Figure 5.8: Same as for figure 5.6a, but for larger increments $\Delta_{\text{acq}} \in [5, 10, 15, 30]$ (a–d). The very same effects are observed for the EIK approach (not shown here).

a numerical strategy based on this common-shot configuration, combining the inverse approaches with asymptotic approximations.

5.3 An asymptotic Born inverse preconditioner for the common-shot iterative migration

Li and Chauris (2018) recently proposed to extend the IVA principles to the common-shot configuration. Compared to the subsurface-offset extension, the inversion formula in this case is not easy to implement, since it implies a division by a Green's function (deconvolution). The computational cost thus increases, and the numerical algorithm must be designed with care to avoid divisions by zeros between Fourier transforms. Fortunately, by comparison, its asymptotic counterpart can be implemented in a straightforward way. Therefore, we propose a numerical strategy that takes benefit from both the common-shot configuration and the asymptotic inversion formula. The principle is to consider the classical wave-equation-based common-shot iterative migration, with an asymptotic preconditioner derived from the formula of Li and Chauris (2018). The preconditioner should provide an improvement to the iterative scheme, by accelerating the convergence. The preliminary results presented in this section are encouraging. Further investigations on this asymptotic preconditioning strategy are necessary for a better understanding of its behavior, especially at large opening angles. Acquisition effects should also be further investigated, as the pseudo-inverses suppose infinite acquisitions.

5.3.1 Theory and implementation

Following Li and Chauris (2018), we rewrite the formulas (5.17), (5.18) and (5.19) in the common-shot formulation, yielding

$$\begin{aligned}
\bar{F}(\xi)(\boldsymbol{s},\boldsymbol{r},\omega) &= (\mathrm{i}\omega)^2\Omega(\omega)\iint \mathrm{d}\boldsymbol{x}\, G_0(\boldsymbol{s},\boldsymbol{x},\omega)\times \\
&\quad \frac{2}{v_0^3(\boldsymbol{x})}\xi(\boldsymbol{x},\boldsymbol{s})G_0(\boldsymbol{x},\boldsymbol{r},\omega), \qquad (5.25)\\
\bar{F}^T(\Delta d)(\boldsymbol{x},\boldsymbol{s}) &= \frac{2}{v_0^3(\boldsymbol{x})}\iint \mathrm{d}\boldsymbol{r}\mathrm{d}\omega\,(\mathrm{i}\omega)^2\Omega^*(\omega)\times \\
&\quad G_0^*(\boldsymbol{s},\boldsymbol{x},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)G_0^*(\boldsymbol{x},\boldsymbol{r},\omega), \qquad (5.26)\\
\bar{F}^\dagger(\Delta d)(\boldsymbol{x},\boldsymbol{s}) &= \frac{8}{v_0^2(\boldsymbol{x})}\int \mathrm{d}\omega\left(|S_0(\boldsymbol{s},\boldsymbol{x},\omega)|^2+\epsilon\right)^{-1}\times \\
&\quad \left(\nabla_{\boldsymbol{x}}S_0^*(\boldsymbol{s},\boldsymbol{x},\omega)\cdot\nabla_{\boldsymbol{x}}R_0(\boldsymbol{s},\boldsymbol{x},\omega)+\left(\frac{\mathrm{i}\omega}{v_0(\boldsymbol{x})}\right)^2 S_0^*(\boldsymbol{s},\boldsymbol{x},\omega)R_0(\boldsymbol{s},\boldsymbol{x},\omega)\right), \qquad (5.27)
\end{aligned}$$

where the symbol ϵ denotes a stabilization parameter to avoid divisions by zero in equation (5.27). The forward and backward propagated wavefields are defined as

$$\begin{aligned}
S_0(\boldsymbol{s},\boldsymbol{x},\omega) &= (\mathrm{i}\omega)^3 G_0(\boldsymbol{s},\boldsymbol{x},\omega)\Omega(\omega), \qquad (5.28)\\
R_0(\boldsymbol{s},\boldsymbol{x},\omega) &= \int \mathrm{d}\boldsymbol{r}\,\partial_{r_z}G_0^*(\boldsymbol{r},\boldsymbol{x},\omega)\Delta d(\boldsymbol{s},\boldsymbol{r},\omega), \qquad (5.29)
\end{aligned}$$

where ∂_{r_z} holds for the vertical derivative of the Green's function at the receiver position. Note that the above operator formulas are very similar to the one given by Qin et al. (2015), and the respective recipes for their derivation from the purely asymptotic inversion formula from Beylkin (1985) and Bleistein (1987) are the same.

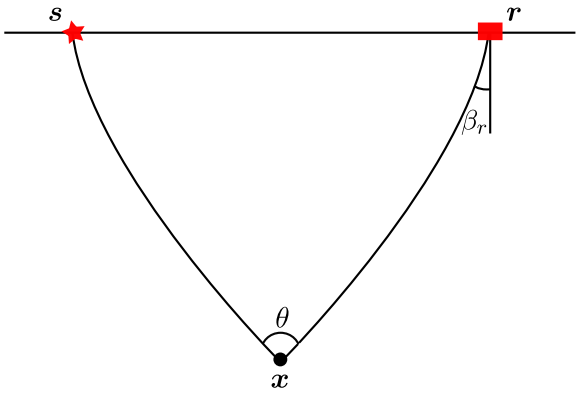


Figure 5.9: Illustration of the quantities involved in the asymptotic inverse formula (5.30). $\beta_{\boldsymbol{r}}$ is the take-off angle at the receiver position, and θ is the opening angle. Note that there is no symmetry between source and receiver side in the common-shot configuration.

The main differences between the subsurface-offset adjoint and inverse operators and the common-shot adjoint and inverse operators are the absence of integral over sources in the latter, as well as a division by a squared wavefield in the common-shot inverse. This division is not simple at all in terms of numerical implementation.

Note that in equation (5.28), there is no derivative with respect to the vertical source position as there is no integration over sources. For that reason, the determinant related to the change of variables from $(\boldsymbol{r}, \omega)$ to (x, z) cannot compensate for the geometrical spreading associated with the source side: a division by $S_0(\boldsymbol{s}, \boldsymbol{x}, \omega)$ is needed in the inverse expression (5.27).

Therefore, we might be interested in using an asymptotic formulation of these operators, which is easier to implement, as we will see. More precisely, in the common-shot approach, we propose to use the wave-equation-based modeling and adjoint operators for a classical iterative migration, with an additional preconditioner, based on the asymptotic, Eikonal-based, formulation of the inverse and its adjoint. This preconditioning strategy should yield a faster convergence, i.e. a faster decrease of the cost function in the iterative migration. The asymptotic inverse operator can be retrieved from the wave-equation formulation (5.27), by replacing the spatial gradients of the Green's functions by a multiplication by $\mathrm{i}\omega\boldsymbol{p}$. A few calculus yields

$$\bar{F}^{\dagger}(\Delta d)(\boldsymbol{x}, \boldsymbol{s}) = -\frac{8}{v_0^2(\boldsymbol{x})} \iint \mathrm{d}\boldsymbol{r}\mathrm{d}\omega \frac{\cos\beta_{\boldsymbol{r}}}{v_{\boldsymbol{r}}} \cos^2(\frac{\theta}{2}) \frac{\Delta d(\boldsymbol{s}, \boldsymbol{r}, \omega)}{\Omega(\omega)} \frac{G_0^*(\boldsymbol{r}, \boldsymbol{x}, \omega)}{G_0(\boldsymbol{s}, \boldsymbol{x}, \omega)}, \tag{5.30}$$

where $v_{\boldsymbol{r}}$ is the background velocity at the receiver position, $\beta_{\boldsymbol{r}}$ is the take-off angle at the receiver position and θ is the opening angle (see figure 5.9). In the asymptotic inverse formulation (5.30), the same kinds of quantities as for the subsurface-offset formulation (5.24) are involved: traveltimes and amplitudes for the computation of asymptotic Green's functions, angles related to the spatial derivatives of traveltimes at the image point, and take-off angles.

However, note that there is no more symmetry between source and receiver side. Take-off angles are only required at the receiver positions. Another difference lies in the angles needed at the image point: in the subsurface-offset case, the vertical derivatives yielded a weight implying the sum of the cosines from source and receiver side (through the sum of vertical components of the slowness vector), while in this common-shot case, we need the cosine of the difference of the angles (opening angle θ). More details on the calculus are given in Li and Chauris (2018). Note also that in this common-shot

formulation, we need to extract both vertical and horizontal components of the slowness vector $\boldsymbol{p}$, while for the subsurface-offset configuration, only p_z was required.

When replacing the Green's functions by their explicit asymptotic expression in the 2D case, equation (5.30) becomes

$$\bar{F}^\dagger(\Delta d)(\boldsymbol{x},\boldsymbol{s}) = -\frac{8}{v_0^2(\boldsymbol{x})}\iint \mathrm{d}\boldsymbol{r}\mathrm{d}\omega \frac{\cos\beta_{\boldsymbol{r}}}{v_{\boldsymbol{r}}}\cos^2(\frac{\theta}{2})\Delta d(\boldsymbol{s},\boldsymbol{r},\omega)\mathrm{i}\Omega^\dagger(\omega)\frac{A(\boldsymbol{x},\boldsymbol{r})}{A(\boldsymbol{s},\boldsymbol{x})}\mathrm{e}^{-\mathrm{i}\omega(\tau(\boldsymbol{r},\boldsymbol{x})+\tau(\boldsymbol{s},\boldsymbol{x}))}, \tag{5.31}$$

where $\Omega^\dagger = \Omega^*/\|\Omega\|^2$ is the inverse of the seismic wavelet. In the preconditioning procedure, we need the adjoint of (5.31), which writes

$$(\bar{F}^\dagger(\Delta d))^T(\xi)(\boldsymbol{s},\boldsymbol{r},\omega) = \mathrm{i}\Omega^{\dagger *}(\omega)\int \mathrm{d}\boldsymbol{x}\frac{8}{v_0^2(\boldsymbol{x})}\frac{\cos\beta_{\boldsymbol{r}}}{v_{\boldsymbol{r}}}\cos^2(\frac{\theta}{2})\xi(\boldsymbol{x},\boldsymbol{s})\frac{A(\boldsymbol{x},\boldsymbol{r})}{A(\boldsymbol{s},\boldsymbol{x})}\mathrm{e}^{\mathrm{i}\omega(\tau(\boldsymbol{r},\boldsymbol{x})+\tau(\boldsymbol{s},\boldsymbol{x}))}. \tag{5.32}$$

The numerical strategy is then the following. We use the wave-equation based modeling and adjoint operators (5.25) and (5.26) in a linear conjugate gradient method for the iterative migration, and we add a preconditioner to this iterative method using the asymptotic inverse operator (5.31) and its adjoint (5.32). A regularization term is added to the misfit function (5.12), yielding the new misfit function

$$\mathcal{C}[\delta m] = \frac{1}{2}\|d_{\mathrm{obs}}(\boldsymbol{s},\boldsymbol{r},\omega) - d_{\mathrm{cal}}[\delta m](\boldsymbol{s},\boldsymbol{r},\omega)\|^2 + \frac{\lambda}{2}\|\nabla_{\boldsymbol{x}}\xi(\boldsymbol{x},\boldsymbol{s})\|^2. \tag{5.33}$$

The parameter λ is computed as follows:

$$\lambda = \alpha\,\frac{\|d_{\mathrm{obs}}(\boldsymbol{s},\boldsymbol{r},\omega)\|^2}{\|\nabla_{\boldsymbol{x}}\delta m_1(\boldsymbol{x},\boldsymbol{s})\|^2}, \tag{5.34}$$

where $\delta m_1(\boldsymbol{x},\boldsymbol{s})$ denotes the velocity perturbation obtained after the first iteration, and α is a chosen parameter. If $\alpha = 1$, the two terms in the misfit function (5.33) have an equivalent weight. Since we want the regularization to be small, we take $\alpha = 10^{-3}$ in the numerical illustrations.

In the next section, I present two numerical illustrations which exhibit the positive effect of the preconditioner on the convergence of the iterative migration.

5.3.2 Numerical results

5.3.2.1 Homogeneous background model with line reflectors

The first numerical test is performed with a homogeneous background velocity $v_0(\boldsymbol{x}) = 3000$ m/s. The physical domain is 1200 m wide and 720 m deep. The finite-difference grid has 201 points in the horizontal direction and 121 points in the vertical direction. This discretization is performed so that sources and receivers, located at the surface along the entire profile, are distributed every two grid points, thus every 12 m. The receiver positions are symmetrical with respect to the source position, with offsets from -600 to $+600$ m. The true squared slowness perturbation consists of five line reflectors, as shown in figure 5.10. Tapers are applied at the domain boundaries in order to avoid spurious reflections at the edges. The source signal is a Ricker signal with a maximum frequency $f_{\mathrm{max}} = 40$ Hz.

The observed data are computed for each shot using the wave-equation-based Born modeling operator (5.25), and plotted at the top of figure 5.13 for a central source position.

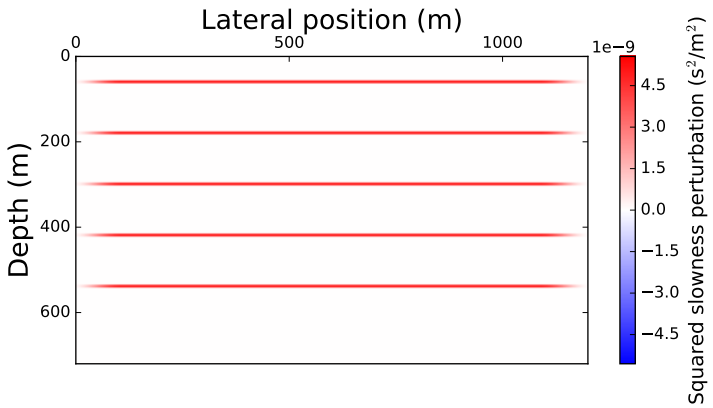


Figure 5.10: True squared slowness perturbation: five line reflectors at depths $z = 60, 180, 300, 420, 540$ m with a squared slowness perturbation $\delta s = 6.10^{-9}$ s^2/m^2.

Five linear conjugate gradient iterations are performed, either with or without the asymptotic preconditioner, in the true homogeneous background model. The asymptotic quantities necessary for the preconditioning with the operators (5.31) and (5.32) are computed with the FSM-DG/FSM-FV solvers, in a 201×121 regular mesh. Again, the center of each cell of the mesh coincides with a point of the FD grid. However, note that in a homogeneous background model, they could be directly obtained with closed-form formulas.

The images (squared slowness perturbation fields) obtained after iterations 1 to 4 are displayed in figure 5.11. Note that iteration 0 is not displayed because there is no migration performed at this step, but only the true data modeling and a measure of the misfit, with a velocity perturbation equal to zero. The reconstruction of the image at depth is drastically improved when the preconditioner is used, and already in the first image. The balance of energy in depth after iteration 1 is notably improved with the preconditioner. This is due to the compensation for the geometrical spreading and for the illumination brought by the pseudo-inverse. Moreover, the sharpness of the reflector lines is better retrieved with the preconditioner, thanks to the seismic wavelet deconvolution included inside the pseudo-inverse formula.

The decrease of the misfit among iterations also highlights the benefit of the preconditioning strategy. This decrease is higher when the preconditioner is applied, as shown in figure 5.12. Finally, after five iterations, the reconstructed data and the data residuals are displayed in figure 5.13. The residuals corresponding to reflectors at depth are strongly reduced when the preconditioner is applied. For the classical approach, the data residual are evenly distributed in the data domain (figure 5.13 bottom left), whereas the residuals with the preconditioner are mainly related to the large angles and to the maximum offsets.

On this simple example, the preconditioning strategy yields a strong improvement of the image as well as an acceleration of the misfit decrease. More work is needed to know if a mute should be applied at the large angles where the direct arrival would be superimposed onto the reflected waves. In order to take benefit from the FSM-DG/FSM-FV solvers, we next perform another numerical illustration with an inhomogeneous background model.

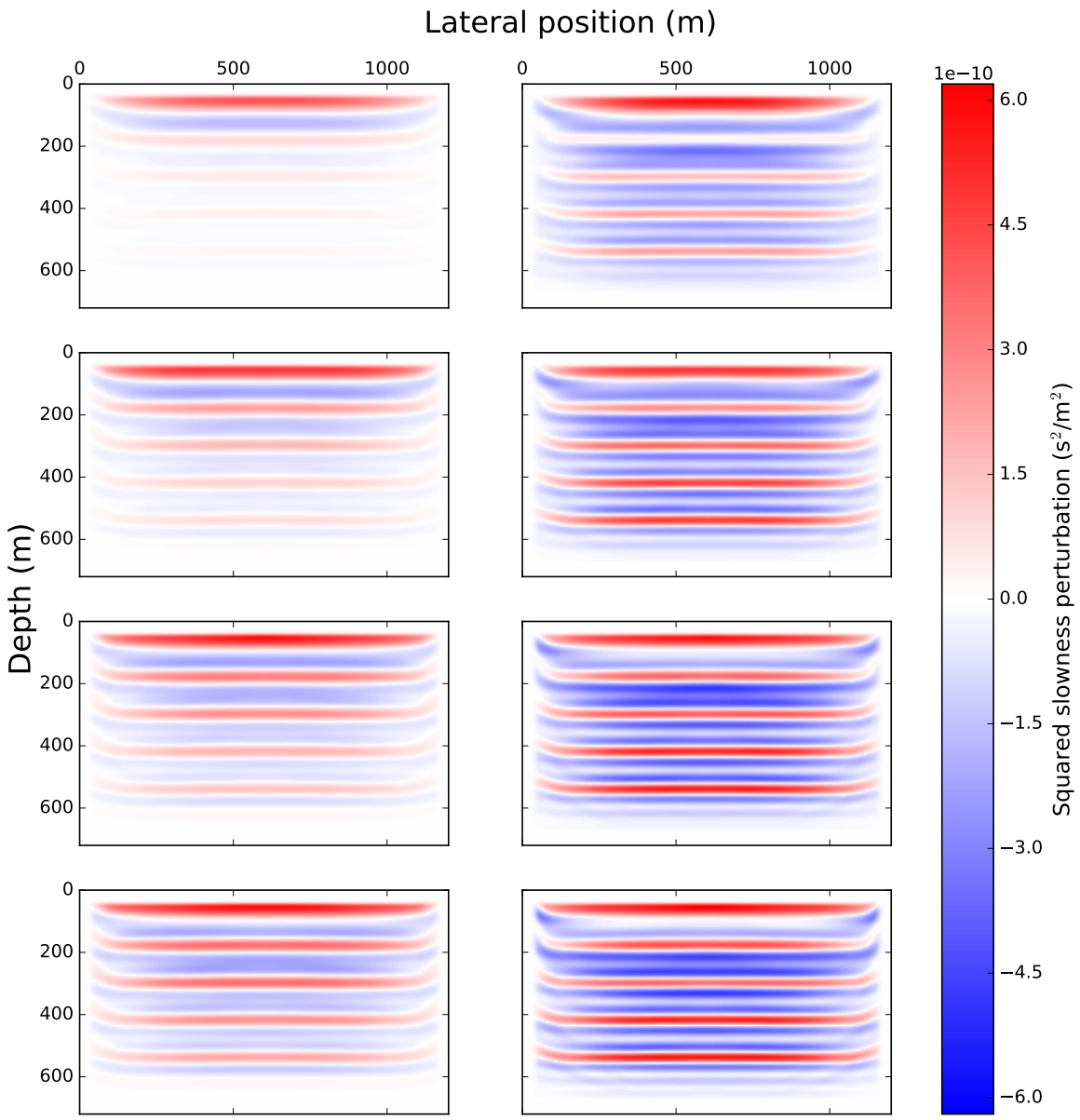


Figure 5.11: From top to bottom: images obtained after iterations 1 to 4, by summing over all shots, in the first numerical example. Left: standard iterative migration, right: iterative migration with asymptotic preconditioning. Note the balance of energy along the depth axis after iteration 1 (top images), and the sharpness of the reflectors when the preconditioner is applied.

5.3.2.2 Marmousi model

For this second numerical test, we use the Marmousi model (Bourgeois et al., 1991), in a physical domain of 2760 m width and 900 m depth. The velocity model is smoothed, and the difference between the original model and the smooth model is used as the true model perturbation. Tapers are applied to the model perturbation at the domain boundaries in order to avoid spurious reflections at the edges. The resulting smooth background model and squared slowness perturbation are displayed in figure 5.14. The

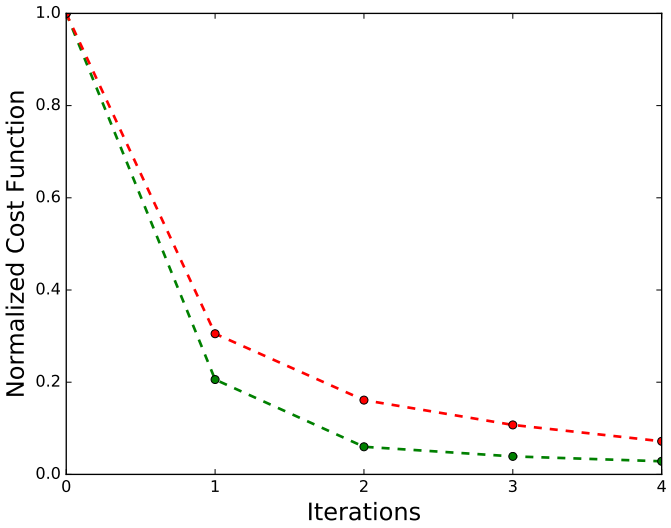


Figure 5.12: Misfit function value among iterations in the first numerical example. Red: standard iterative migration, green: iterative migration with asymptotic preconditioning.

finite-difference grid has 151 points in the horizontal direction and 461 points in the vertical direction. This discretization is performed so that sources and receivers, located at the surface along the entire profile, are distributed every two grid points, thus every 12 m. The receiver positions are symmetrical with respect to the source position, with offsets from -600 to $+600$ m. The source signal is a Ricker signal with a maximum frequency $f_{\max} = 7.5$ Hz.

The observed data are computed for each shot using the wave-equation-based Born modeling operator (5.25), and plotted at the top of figure 5.19 for a central source position.

Five linear conjugate gradient iterations are performed, either with or without the asymptotic preconditioner, in the true background model, which, here again, is assumed to be known. The asymptotic quantities necessary for the preconditioning with operators (5.31) and (5.32) are computed with the FSM-DG/FSM-FV solvers, in a 151×461 regular mesh. Again, the center of each cell of the mesh coincides with a point of the FD grid. The complexity of the background model yields several triplications of the wavefield. This is visible in figure 5.15, which displays the traveltime and amplitude maps computed for sources located at $x = 0$ and $x = 1500$ m. Note that the maps have been computed in the whole model, while the offsets used in this experiments are limited to ± 600 m. As a consequence, the triplications should have only a small effect in this example, since most of them occur out of the limited-offset region. Moreover, the computation of the whole maps are not required, so that a significant part of the computational cost could be saved by reducing the size of the maps to the required region.

The images (squared slowness perturbation fields) obtained after iterations 1 to 4 are displayed in figure 5.16, with and without the preconditioner. The differences between both strategies are less visible than in the previous example. However, we note that the magnitude of the perturbation field is slightly better retrieved with the preconditioner. This is visible in the vertical profiles extracted from the images and displayed in figure 5.17.

The decrease of the misfit among iterations highlights the benefit of the preconditioning strategy. This decrease is slightly higher when the preconditioner is applied, as shown in figure 5.18. Finally, after five iterations, the reconstructed data and the data residuals are displayed in figure 5.19. Once

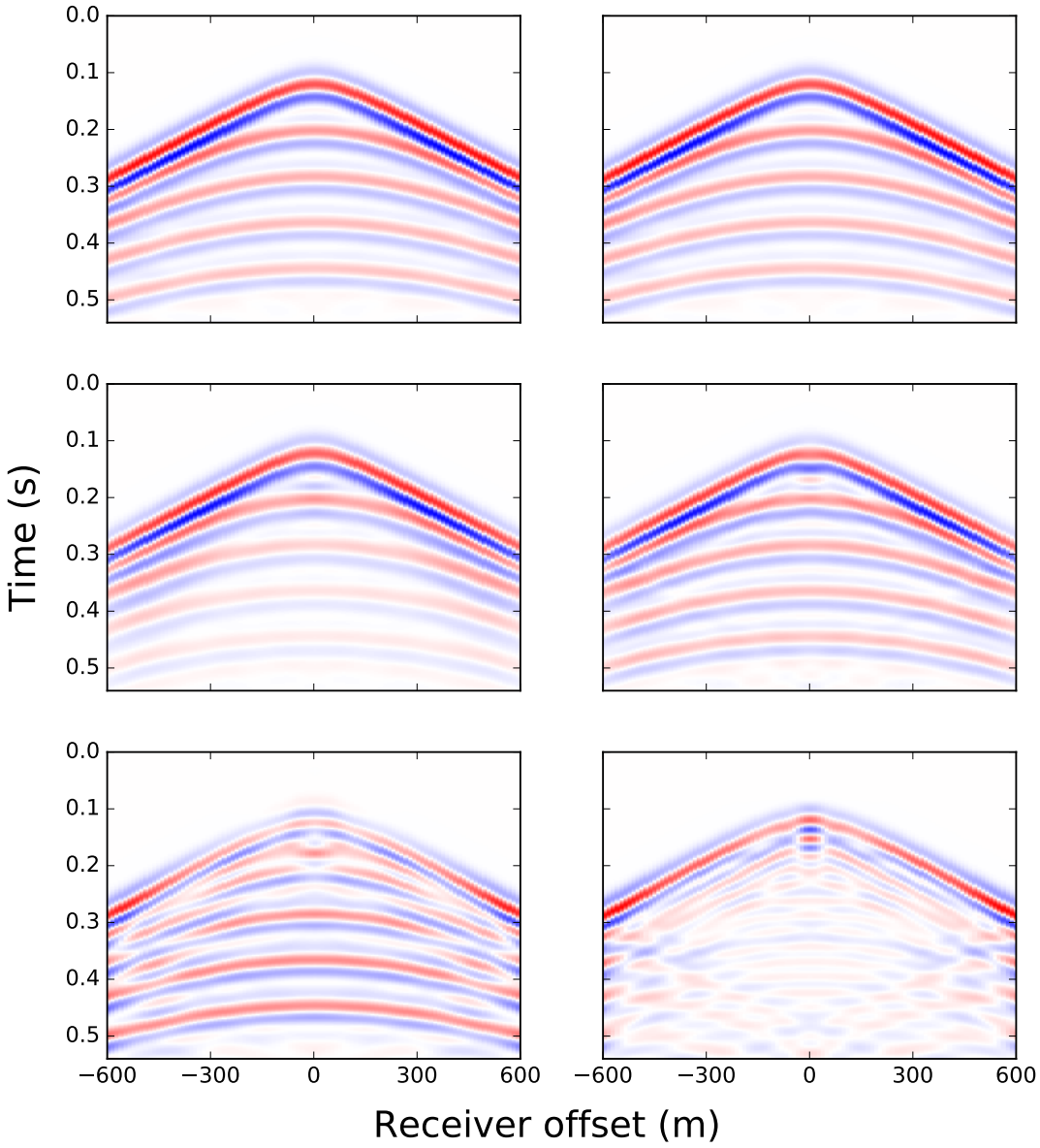


Figure 5.13: Common-shot gathers for a central source in the first numerical example. Left: standard iterative migration, right: iterative migration with asymptotic preconditioning. Top: observed data, middle: data reconstructed after 5 iterations, bottom: data residuals (difference between observed and reconstructed data) after 5 iterations. Top and middle are displayed at same scale, while the data is magnified by a factor 3 for bottom plots.

more, the data residuals have two different distributions (figure 5.19 bottom). If the data residuals were evaluated in the central part of the shot, there would be a significant difference between the standard and preconditioned approach.

The computation is parallelized per source. Without preconditioner, it took approximately 6 hours using 16 cores of one cluster node comprised of twelve-core Intel® Xeon® E5-2695 processors with frequency 2.4 GHz. The computation with the preconditioner took approximately 11.5 hours using 16

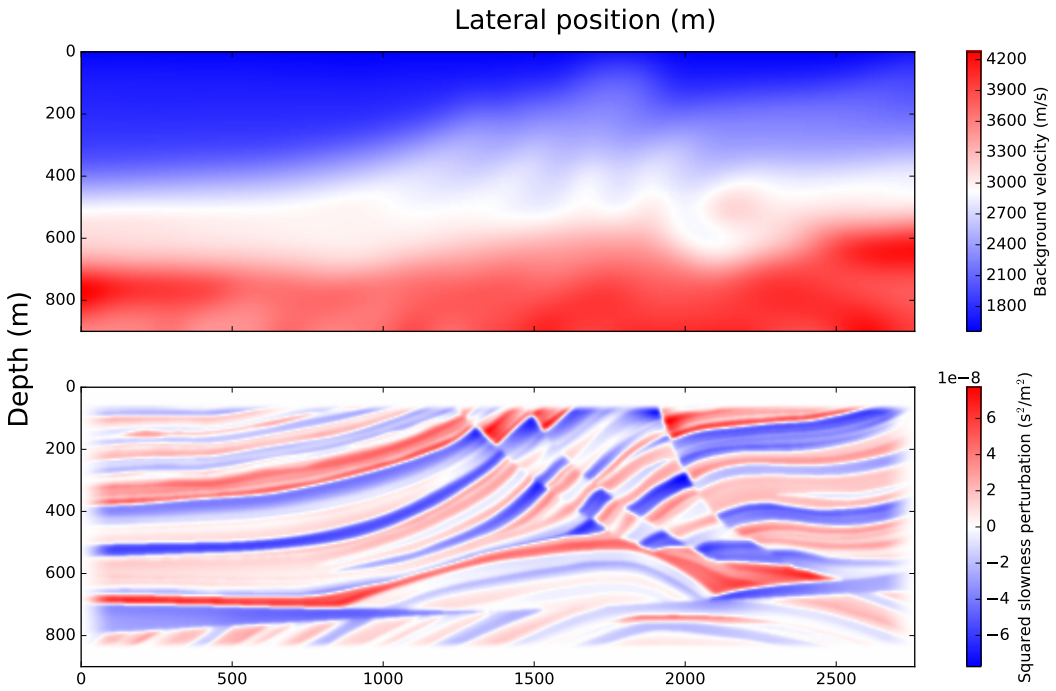
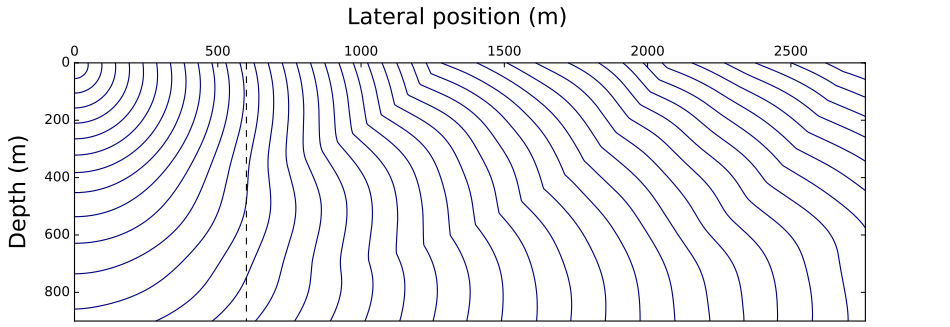


Figure 5.14: The Marmousi true model. Top: background velocity, bottom: squared slowness perturbation.

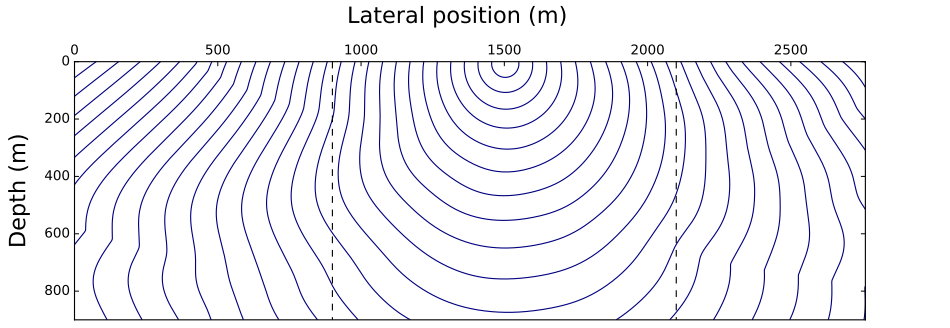
cores of one cluster node comprised of eight-core Intel® Xeon® E5-2680 processors with frequency 2.5 GHz. Note again that these are only orders of magnitude. Here again, the asymptotic computations could be optimized a lot in order to save computational cost.

5.3.3 Conclusion

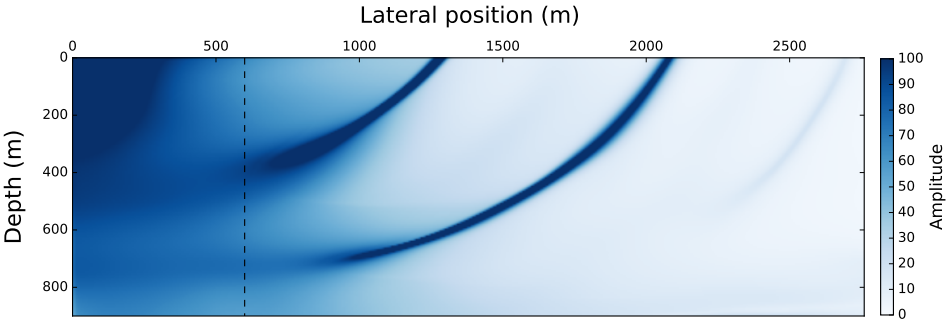
The two numerical examples illustrate the benefit of the proposed preconditioning strategy. In the Marmousi example, the preconditioning strategy yields an improvement of the image as well as an acceleration of the misfit decrease. However, the benefit seems weaker than in the first example. The first reason for that could be related to the triplications in the wavefield, that are not modeled by the first-arrival-only asymptotic quantities. However, as mentioned before and shown in figure 5.15, the resulting triplications occur only for large offsets, while the offsets are limited to ± 600 m in the example. Many other parameters are involved, so that further investigations are necessary. For instance, the price of going to higher signal frequency should be paid in order to get closer to the high-frequency hypothesis. The maximum receiver offsets to be considered for each source should affect the results and should therefore be also considered, because we have seen that the preconditioner is less efficient for wide opening angles.



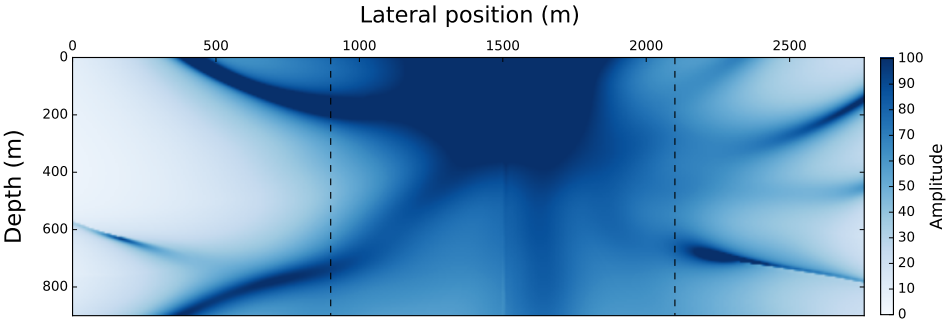
(a)



(b)



(c)



(d)

Figure 5.15: Marmousi model. Traveltime isocontours and amplitude maps for sources located at $x = 0$ m (a and c), and at $x = 1500$ m (b and d).

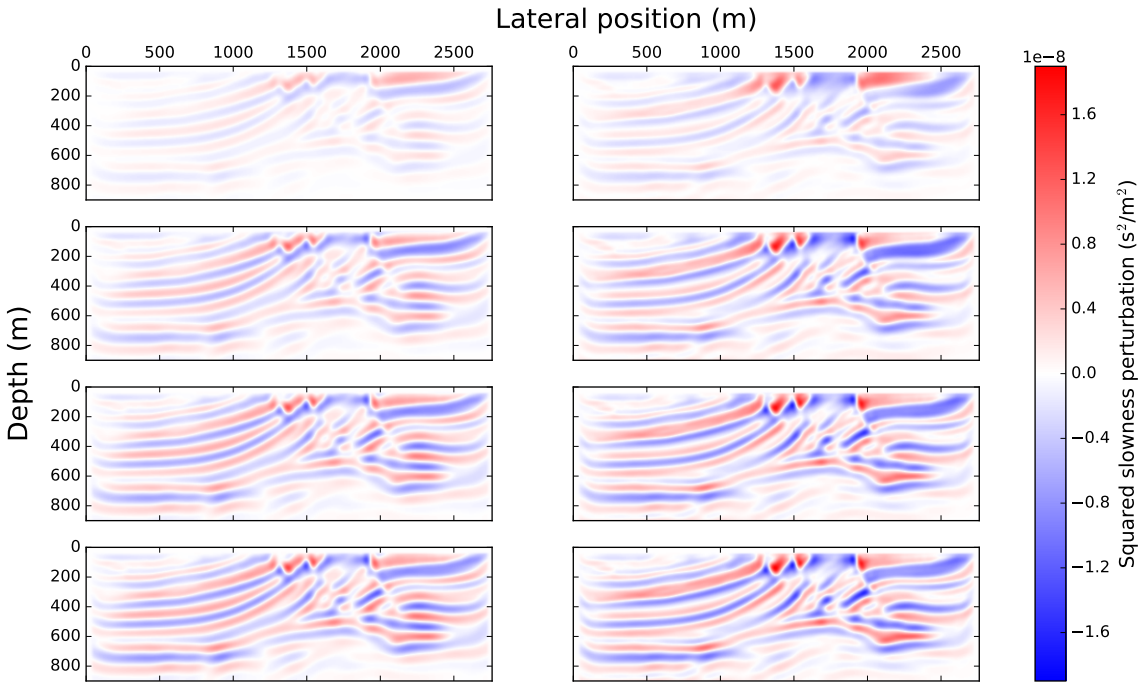


Figure 5.16: From top to bottom: images obtained after iterations 1 to 4, by summing over all shots, in the Marmousi example. Left: standard iterative migration, right: iterative migration with asymptotic preconditioning.

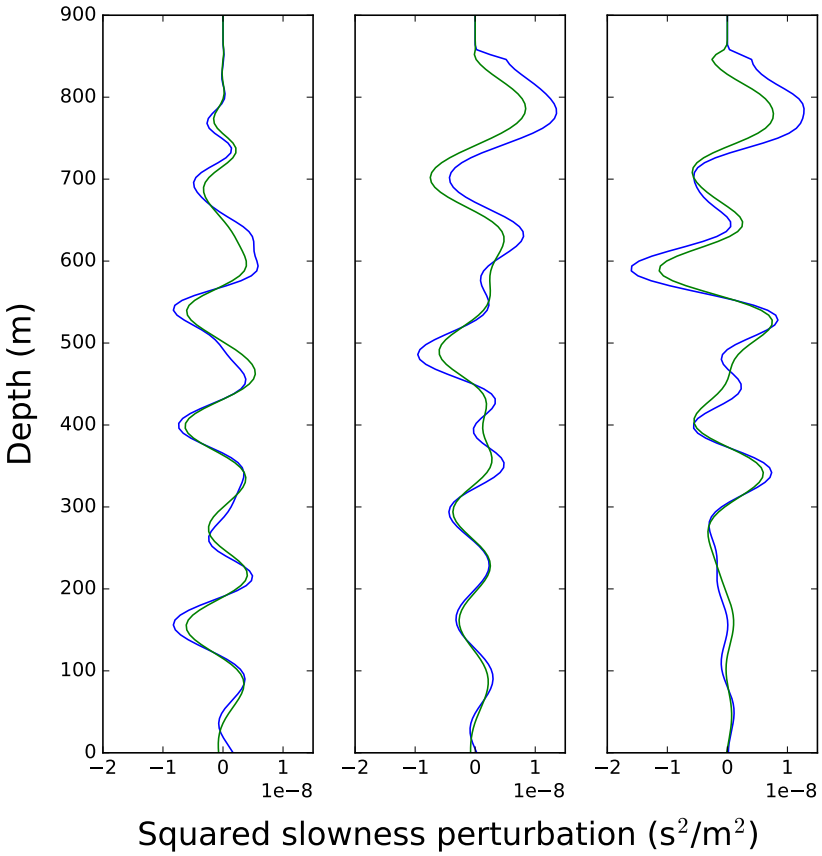


Figure 5.17: Vertical profiles of squared slowness perturbation extracted after five iterations, in the Marmousi example, at $x = 690, 1410, 1980$ m, from left to right. Green: standard iterative migration, blue: iterative migration with asymptotic preconditioning. Note the higher magnitude of the perturbation when the preconditioner is applied.

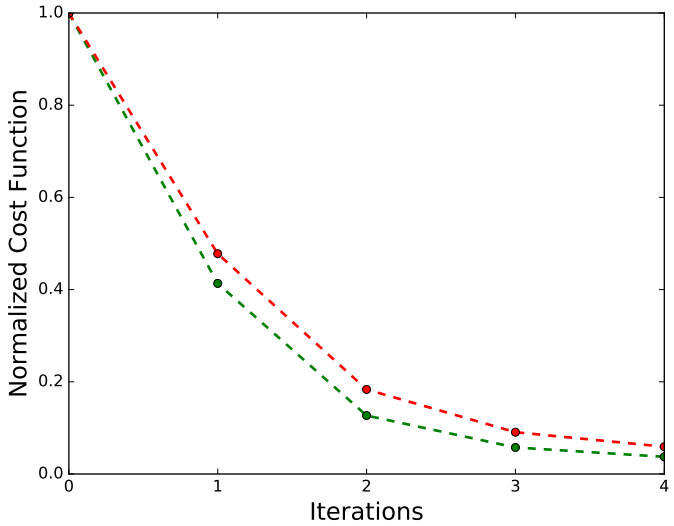


Figure 5.18: Misfit function value among iterations, in the Marmousi example. Red: standard iterative migration, green: iterative migration with asymptotic preconditioning.

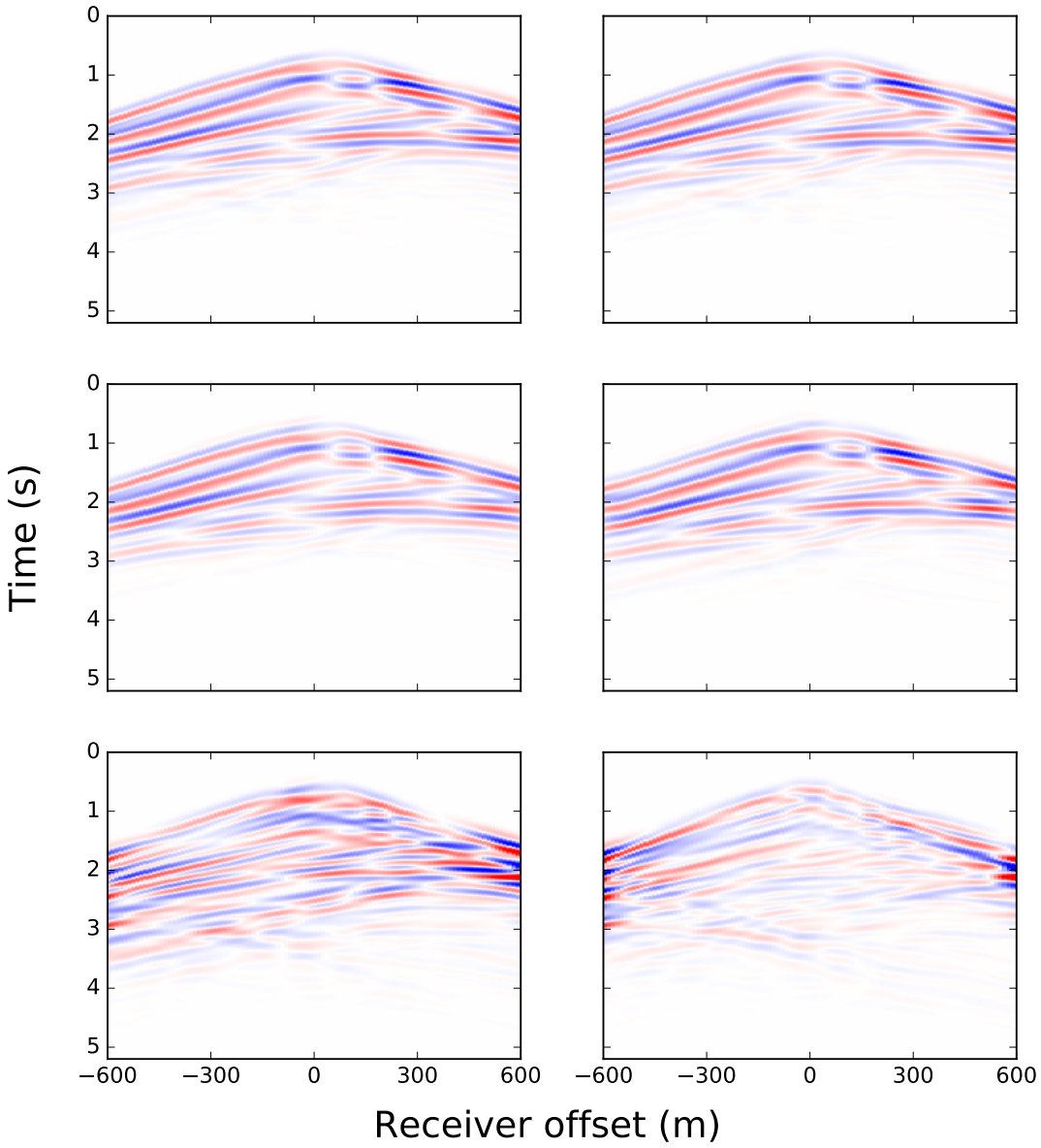


Figure 5.19: Common-shot gathers for a central source in the Marmousi example. Left: standard iterative migration, right: iterative migration with asymptotic preconditioning. Top: observed data, middle: data reconstructed after 5 iterations, bottom: data residuals (difference between observed and reconstructed data) after 5 iterations. Top and middle are displayed at same scale, while the data is magnified by a factor 5 for bottom plots.

5.4 Conclusion

In this chapter, I have presented results obtained in the Born inversion framework. The first objective, building on the previous chapters, was to exhibit an application of the asymptotic solvers in a seismic imaging context. In particular, I have chosen applications that require not only the traveltime, but also the amplitude, as well as angle information, in order to combine all these tools.

Moreover, this was an opportunity for developing new imaging strategies based on recent advances in the field. The preliminary results that we obtained are encouraging, so that further investigation would be desirable, together with numerical developments in order to make the algorithms more efficient.

The asymptotic approaches result in the computation of maps of quantities such as traveltime, amplitude, and take-off angle. These computations take advantage from the separation, through the ray ansatz, between the highly-oscillating part of the wavefield and the smoothly varying traveltime and amplitude. The advantage of a slowly varying quantity (traveltime) compared to the highly oscillating wavefield makes efficient decimations/interpolations possible over sparse grids for storage, which might be of great interest when working with large models and datasets (Mendes, 2000; Vanelle and Gajewski, 2002; Alkhalifah, 2011). Thus, for large 3D problems, all the maps could be either computed on-the-fly, or interpolated from a decimated reference set, during the inversion and modeling steps. An extension of this interpolation is proposed in Li and Fomel (2013), which consists of the derivative of the traveltime with respect to the source location. These derivatives can be accessed by solving a first-order Hamilton–Jacobi PDE. Using this advanced interpolation would be advantageous in terms of computational cost. Keeping a high accuracy in the traveltime is made possible, even when decimating the reference set of traveltime maps.

In all the numerical illustrations, the observed data are modeled in the same physics (Born approximation) as the imaging in the WEQ approach (section 5.2) or in the case without preconditioner (section 5.3). In future investigations, other approaches could be considered so as to perform fair comparisons out of this inverse crime configuration. Noise could be added to the data. Cases where the exact background model is not known should be considered, as well as a combination with MVA/IVA. Eventually, if the interest of these approaches is confirmed, the algorithms could be applied to real data.

Conclusions and perspectives

Conclusions

Using a high-frequency approximation for the numerical modeling of seismic wave propagation is computationally advantageous when hundreds, thousands, or more of wavelengths have to be propagated. In this context, the Eulerian point of view is particularly suitable in practice for a large variety of applications, such as seismic imaging and inversion. The HF approximation yields a set of partial differential equations, that are derived in chapter 1, for the computation of traveltime, amplitude, take-off angle, and adjoint-state variable. The Eikonal equation, for traveltime computation, is a non-linear Hamilton–Jacobi equation. It admits a particular type of generalized solutions, namely the viscosity solution, which corresponds physically to the first arrival traveltime.

It is possible to design numerical algorithms that converge towards this viscosity solution. While the majority of them have been developed in a finite-difference framework, I propose in chapter 2 a discontinuous Galerkin framework. Coming from an applied mathematics context and appropriate for the whole Hamilton–Jacobi family, I propose several adaptations to the geophysical problem. For that purpose, I implement accurate point-source handling, as well as appropriate initial and boundary conditions. The boundary conditions result in a new term in the DG formulation. I also select the most efficient Hamiltonians for isotropic and 2D tilted transversely isotropic media. I exhibit the main advantages of this new Eikonal solver for seismic traveltime computation. Complex heterogeneous media are correctly handled. The general HJ framework is appropriate for any kind of anisotropy. Complex geometries such as non-flat topographies can be efficiently handled. Finally, it achieves an unprecedented accuracy, when compared with existing FD solvers.

The Runge–Kutta DG method is very efficient for time-dependent problems. However, I have used it for a steady-state computation, and I have found that in such a case, the algorithmic complexity is in $\mathcal{O}(N^3/2)$, where N is the total number of degrees of freedom. This is higher than classical Eikonal solvers, which reach a linear complexity in $\mathcal{O}(N)$. Therefore, it was necessary to find a way to accelerate the steady-state computation.

In chapter 3, I present the fast-sweeping global procedure that I have adapted to the DG scheme. Inspired from standard fast-sweeping methods classically built on FD schemes, it consists of block-Gauss–Seidel iterations in the DG formulation. Solving the blocks is performed through a local solver that updates the solution inside a cell of the mesh. This local solver is based on a first-order explicit Euler integration. I have designed an efficient degenerate local solver in order to overcome some transient configurations, in which a local convergence may not be reached due to the nonlinearity of the problem and causality considerations. Based on plane-wave propagation with suitable causality orientation, this degenerate local solver is a key ingredient to ensure the robustness of the resulting new FSM-DG method.

The FSM-DG method is shown to be highly accurate and computationally efficient, yielding a linear complexity in $\mathcal{O}(N)$, and a drastic decrease of the computational cost. It is therefore possible to consider computations in three dimensions, which I demonstrate in isotropic as well as tilted orthorhombic media.

In chapter 4, I capitalize on this new FSM-DG Eikonal solver to derive others solvers for take-off angle, amplitude, and adjoint-state variable, in a unified, consistent framework. The equation for take-off angle is a linear Hamilton–Jacobi equation. It is therefore easy to use the same FSM-DG solver for take-off angle as for traveltime. The equations for amplitude and adjoint-state variable are linear transport equations. I take benefit from the conservative formulation of these equations and the Green–Ostrogradsky formula to design finite-volume solvers which involve only the first-order derivatives of the traveltime. The key ingredients of all these solvers are the common discretization, the extraction of consistent quantities from the DG Eikonal solver for traveltime derivative values, the DG/FV local solvers, and the common FSM global procedure.

As a result, a whole set of consistent solvers is proposed to the geophysical community. These solvers should be useful in a wide range of applications, especially when numerous sources and receivers are involved, when highly accurate traveltimes are needed, and/or when the traveltime derivatives are involved. This is the case for instance in slope tomography and in Born/Kirchhoff imaging algorithms.

An illustration is given in chapter 5, where I use these solvers in an advanced imaging algorithm. Traveltime, amplitude, and take-off angle are required in the asymptotic Born approximate inverse operator. Asymptotic ray+Born imaging schemes are known to be computationally less demanding than wave-equation based schemes. This is due to the computation of smoothly varying quantity instead of full wavefields, and to the decimation of sources/receivers that can reduce the number of fields to precompute thanks to subsequent interpolation. The results shown in chapter 5 are preliminary results, and further investigations are necessary for a better understanding of the imaging behavior and for larger-scale, realistic studies. However, these preliminary results exhibit an improvement of the image as well as an acceleration of the convergence when asymptotic quantities are employed in a preconditioning strategy.

Perspectives

Numerical developments A first range of perspectives that can be drawn from this work is related to further numerical developments in order to improve the solvers in terms of accuracy and efficiency. First, the amplitude computation detailed in section 4.2.1 exhibits artifacts linked to the grid. These artifacts could be treated in several ways, such as by improving the point-source factorization, or by finding more accurate ways to verify the energy conservation. Second, anisotropy has not been accounted for in angle computation (section 4.1) and in amplitude computation (section 4.2.1) because artifacts appear when anisotropy is introduced. These artifacts need to be treated together with a better understanding of the numerical energy propagation in presence of anisotropy. Finally, so far, only prototype solvers have been developed. For real applications, the whole set of solvers presented in this work could be optimized with high-performance computing techniques for the sake of numerical efficiency.

Interest of angles and curvatures The DG polynomial approximation for the traveltime computation offers a direct access to the spatial derivatives of the traveltime, thus to the direction of propagation

of the waves (first-order derivatives) and the local wavefront curvature (second-order derivatives). In the full wavefield, when several phases interfere, it is not possible to separate them, and taking the spatial derivatives of the calculated wavefield does not really make sense physically. Alternatively, angle modeling, by means of solving take-off angle equation or considering first-order traveltime derivatives, gives a physical meaning to the propagation direction of each phase. Indeed, in the case of wave interference, a multivalued computation of asymptotic quantities would be necessary, and would yield an interesting understanding of the wave superposition.

Going up in order with the DG computation, one can also consider second-order derivatives of the traveltime. This would be interesting for applications requiring curvature analysis, such as the CRS analysis (Dell and Gajewski, 2011; Schwarz et al., 2014; Schwarz and Gajewski, 2017), and the focus on diffractions in seismic imaging (Bauer et al., 2017).

Geophysical applications in presence of anisotropy The migration/inversion algorithms presented in chapter 5 could be extended to anisotropic media. To do so, the computation of amplitude and take-off angle maps in presence of anisotropy are required, and the wave-equation-based propagators must handle anisotropy as well.

The adjoint-state method is used in Waheed et al. (2016) for the first-arrival traveltime tomography for anisotropic media. Incorporating the FSM-DG/FSM-FV solvers developed in this thesis inside an adjoint-state anisotropic traveltime tomography algorithm is currently being developed and tested in presence of non-flat topography.

Recent implementations of slope tomography with the adjoint-state method use Eikonal and transport solvers in finite differences for traveltime, slope, and adjoint-state variable (Tavakoli F. et al., 2017; Tavakoli F. et al., 2018; Sambolian et al., 2018). Slopes are required at the surface and at depth. In the FD implementation, these slopes are computed with an additional FD operator. Interestingly, the DG approach developed in this thesis would yield a more accurate way to obtain them. Similar to the example in seismic imaging in chapter 5, slopes at depth can be directly accessed from the traveltime derivatives given by the DG polynomial solution. Slopes at the surface can be computed by solving the take-off angle equation as detailed in chapter 4. In addition, slope tomography could be implemented in presence of non-flat topography using the solvers developed in this thesis.

Multivalued traveltime Triplication of the wavefield occurs even in smooth models, like in the smoothed Marmousi background model in section 5.3.2.2 for instance. The traveltime solver developed in this thesis considers the viscosity solution of the Eikonal equation, namely the first-arrival traveltime only. In presence of triplication, the whole description of the wavefield in an asymptotic sense requires the modeling of all the successive arrivals. In other words, all the branches of the solution have to be modeled. It is well known that multivalued traveltime should be considered in applications such as migration for accurate images (Geoltrain and Brac, 1993). However, it is very challenging to design simple, efficient, and robust algorithms for extracting multivalued traveltime.

In the Lagrangian viewpoint, detailed in section 1.2.1, rays might be traced in all the directions, so that in theory, all the arrivals could be modeled. However, the spatial resolution cannot be controlled in the regions where the rays diverge, and shadow zones are not handled by the standard ray theory. For the reasons mentioned in section 1.2.2, the semi-Lagrangian, wavefront tracking techniques are quite complex in terms of numerical implementation.

Besides, the use of uniform grids in the Eulerian viewpoint is desirable. On the one hand, the multivalued traveltime solution might be partitioned into a set of first-arrival solutions with suitable

boundary conditions. This approach makes use of classical Eikonal solvers and has been proposed by Fatemi et al. (1995), Bevc (1997), Benamou (1999), and Symes and Qian (2003). However, the structure of the wavefield and its multivaluedness are not known a priori, so that the partition of the solution cannot be defined in general cases. On the other hand, one might consider additional dimensions to the standard space.

In the multi-arrival formulation, each point in the phase space is localized by its space coordinate $\boldsymbol{x}$ but also by its momentum $\boldsymbol{p}$, corresponding to the direction of propagation. At a given point $\boldsymbol{x}$ of the standard space, multiple times might be modeled which correspond to multiple momenta $\boldsymbol{p}$. Fomel and Sethian (2012) propose to solve a system of PDEs in reduced phase space. Alternatively, level-set methods (Osher and Sethian, 1988) track the evolution of an interface, and are investigated in Osher et al. (2002) and Hauser et al. (2006). A review can be found in Liu et al. (2006).

These grid-based methods involve grid computations which are generally performed with FD algorithms, although a spectral/DG method is developed in Cockburn et al. (2005). The FSM-DG Eikonal solver presented in this thesis could be implemented and tested in these methods. Moreover, in some seismic imaging applications, it is reasonable to assume the paraxial condition, stating that the rays are oriented in the positive z-direction. In this case, the depth z can be used as a running parameter. An isotropic Hamiltonian in the reduced phase space (x, p_x, z) might write

$$\mathcal{H}(x, p_x, z) = \sqrt{\max\left(s^2(x,z) - p_x^2, s^2(x,z)\cos^2\theta_{\max}\right)}, \tag{5.35}$$

where $s(x,z)$ is the slowness. This Hamiltonian is suitable for rays making an angle $\theta \leq \theta_{\max} < \pi/2$ with the vertical. In this context, the paraxial Eikonal equation states

$$p_z = \mathcal{H}(x, p_x, z). \tag{5.36}$$

This is established in Symes and Qian (2003) and used in Qian and Leung (2004). Interestingly, assimilating z to the pseudo-time ξ used in chapter 2, the RK-DG solver could be used for solving this paraxial Eikonal equation, since it writes like a dynamic Hamilton–Jacobi equation for the traveltime $T(x, p_x, z)$:

$$\partial_z T - \mathcal{H}(x, p_x, z) = 0. \tag{5.37}$$

The 2D space used for the RK-DG computation would then be (x, p_x), and the evolution parameter would be z, resulting in a 2D+time integration problem, instead of a 4D problem in the full phase space. Each Runge–Kutta integration step would correspond to a march in depth, starting from the surface at $z = 0$. I have shown in introduction of chapter 3 that the RK-DG scheme is not efficient for a steady-state computation. However, in this paraxial case, the RK-DG scheme could be of great help for a multivalued traveltime computation.

In the imaging applications like the ones that are described in chapter 5, the paraxial approximation is a good approximation for most cases. Therefore, a paraxial multivalued traveltime solver should improve the image reconstruction in presence of triplication. Considering all the arrivals would allow to conserve the energy and focus it where it should be. This is true in particular when the most energetic arrival does not correspond to the first arrival.

Bibliography

Abma, R., Sun, J., and Bernitsas, N. (1999). Antialiasing methods in kirchhoff migration. *Geophysics*, 64:1783–1792.

Aki, K., Christoffersson, A., and Husebye, E. S. (1977). Determination of the three-dimensional seismic structure of the lithosphere. *Journal of Geophysical Research*, 82(2):277–296.

Aki, K. and Lee, W. (1976). Determination of three-dimentional velocity anomalies under a seismic array using first p-arrival times from local earthquakes. *Journal of Geophysical Research*, 81(4):4381–4399.

Alkhalifah, T. (2000). An acoustic wave equation for anisotropic media. *Geophysics*, 65:1239–1250.

Alkhalifah, T. (2003). An acoustic wave equation for orthorhombic anisotropy. *Geophysics*, 68:1169–1172.

Alkhalifah, T. (2011). Efficient traveltime compression for 3D prestack Kirchhoff migration. *Geophysical Prospecting*, 59(1):1–9.

Araki, T., Enomoto, S., Furuno, K., Gando, Y., et al. (2005). Experimental investigation of geologically produced antineutrinos with KamLAND. *Nature*, 436:499–503.

Babic, V. M. and Buldyrev, V. S. (1991). *Short-wavelength diffraction theory - asymptotic methods*, volume 4. New York, Springer-Verlag, Berlin, Heidelberg.

Babič, V. M. and Kirpičnikova, N. Y. (1979). *The Boundary-Layer Method in Diffraction Problems*, volume 3. Springer-Verlag, Berlin.

Babuska, V. and Cara, M. (1991). *Seismic Anisotropy in the Earth*. Boston: Kluwer Academic Publishers.

Backus, G. E. (1962). Long-wave elastic anisotropy produced by horizontal layering. *Journal Geophysical Research*, 67:4427–4440.

Baina, R., Nguyen, S., Noble, M., and Lambaré, G. (2003). Optimal anti-aliasing for ray-based Kirchhoff depth migration. In *SEG Technical Program Expanded Abstracts 2003*, pages 1130–1133. Society of Exploration Geophysicists.

Bakulin, A., Grechka, V., and Tsvankin, I. (2000). Estimation of fracture parameters from reflection seismic data—Part II: fractured models with orthorhombic symmetry. *Geophysics*, 65(6):1803–1817.

Bauer, A., Schwarz, B., and Gajewski, D. (2017). Utilizing diffractions in wavefront tomography. *Geophysics*, 82(2):R65–R73.

Belayouni, N. (2013). *Nouveaux algorithmes efficaces de modélisation 2D et 3D: Temps des premières arrivées, angles à la source et amplitudes*. PhD thesis, Ecole nationale supérieure des mines de Paris.

Benamou, J.-D. (1999). Direct computation of multivalued phase space solutions for hamilton–jacobi equations. *Communications on Pure and Applied Mathematics*, 52(11):1443–1475.

Benamou, J. D. (2003). An introduction to Eulerian geometrical optics (1992–2002). *Journal of Scientific Computing*, 19(1):63–93.

Bevc, D. (1997). Imaging complex structures with semirecursive Kirchhoff migration. *Geophysics*, 62(2):577–588.

Beylkin, G. (1985). Imaging of discontinuities in the inverse scaterring problem by inversion of a causal generalized Radon transform. *Journal of Mathematical Physics*, 26:99–108.

Beylkin, G. and Burridge, R. (1990). Linearized inverse scattering problems in acoustics and elasticity. *Wave motion*, 12:15–52.

Billette, F. and Lambaré, G. (1998). Velocity macro-model estimation from seismic reflection data by stereotomography. *Geophysical Journal International*, 135(2):671–680.

Biondi, B. (1992). Solving the frequency dependent eikonal equation. In *Expanded Abstracts*, pages 1315–1319. Soc. Expl. Geophys.

Biondi, B. and Symes, W. (2004). Angle-domain common-image gathers for migration velocity analysis by wavefield-continuation imaging. *Geophysics*, 69(5):1283–1298.

Bishop, T. N., Bube, K. P., Cutler, R. T., Langan, R. T., Love, P. L., Resnick, J. R., Shuey, R. T., and Spinder, D. A. (1985). Tomographic determination of velocity and depth in laterally varying media. *Geophysics*, 50:903–923.

Bleistein, N. (1987). On the imaging of reflectors in the Earth. *Geophysics*, 52(7):931–942.

Bonnasse-Gahot, M., Calandra, H., Diaz, J., and Lanteri, S. (2018). Hybridizable discontinuous Galerkin method for the 2-D frequency-domain elastic wave equations. *Geophysical Journal International*, 213(1):637–659.

Boué, M. and Dupuis, P. (1999). Markov chain approximations for deterministic control problems with affine dynamics and quadratic cost in the control. *SIAM Journal on Numerical Analysis*, 36(3):667–695.

Bourgeois, A., Bourget, M., Lailly, P., Poulet, M., Ricarte, P., and Versteeg, R. (1991). Marmousi, model and data. In *The Marmousi Experience*, pages 5–16. Eur. Ass. Expl. Geophys.

Brossier, R. (2009). *Imagerie sismique à deux dimensions des milieux visco-élastiques par inversion des formes d'onde: développements méthodologiques et applications.* PhD thesis, Université de Nice-Sophia-Antipolis.

Buske, S. (1999). Three-dimensional pre-stack Kirchhoff migration of deep seismic reflection data. *Geophysical Journal International*, 137(1):243–260.

Buske, S. and Kästner, U. (2004). Efficient and accurate computation of seismic traveltimes and amplitudes. *Geophysical Prospecting*, 52:313–322.

Cao, S. and Greenhalgh, S. (1994). Finite-difference solution of the eikonal equation using an efficient, first-arrival, wavefront tracking scheme. *Geophysics*, 59:632–643.

Capdeville, Y., Guillot, L., and Marigo, J.-J. (2010). 2-D non-periodic homogenization to upscale elastic media for P-SV waves. *Geophysical Journal International*, 182:903–922.

Carcione, J., Herman, G., and ten Kroode, A. (2002). Seismic modeling. *Geophysics*, 67(4):1304–1325.

Carcione, J., Kosloff, D., and Kosloff, R. (1988). Wave-propagation simulation in an elastic anisotropic (transversely isotropic) solid. *Quarterly Journal of Mechanics and Applied Mathematics*, 41(3):319–345.

Carrion, P., Boehm, G., Marchetti, A., Pettenati, F., and Vesnaver, A. (1993). Reconstruction of lateral gradients from reflection tomography. *Journal of Seismic Exploration*, pages 55–67.

Červený, V. (2001). *Seismic Ray Theory*. Cambridge University Press, Cambridge.

Červený, V., Klimeš, V., and Pšenčik, I. (1988). Complete seismic-ray tracing in three-dimensional structures. In Doornbos, D. J., editor, *Seismological algorithms*, pages 89–168. Academic Press, New York.

Chapman, C. (2004). *Fundamentals of seismic waves propagation*. Cambridge University Press, Cambridge, England.

Chapman, C. H. (1985). Ray theory and its extensions: WKBJ and Maslov seismograms. *Journal of Geophysics*, 58:27–43.

Chauris, H. and Cocher, E. (2017). From migration to inversion velocity analysis. *Geophysics*, 82(3):S207–S223.

Chauris, H. and Cocher, E. (2018). Review of different expressions for the extended Born approximate inverse operator. In *Expanded Abstracts, 80^{th} Annual EAGE Meeting*, pages accepted to Workshop 07 – Seismic Imaging with Ray and Waves – Where do we stand? Part II: Imaging.

Cheng, Y. and Shu, C.-W. (2007). A discontinuous Galerkin finite element method for directly solving the Hamilton-Jacobi equations. *Journal of Computational Physics*, 223(1):398 – 415.

Cheng, Y. and Wang, Z. (2014). A new discontinuous Galerkin finite element method for directly solving the Hamilton–Jacobi equations. *Journal of Computational Physics*, 268:134–153.

Claerbout, J. (1971). Towards a unified theory of reflector mapping. *Geophysics*, 36:467–481.

Cockburn, B. (2003). Discontinuous galerkin methods. *Journal of Applied Mathematics and Mechanics*, 83:731–754.

Cockburn, B., Karniadakis, G., and Shu, C.-W. (2000). The development of discontinuous Galerkin methods. In Cockburn, B., Karniadakis, G., and Shu, C.-W., editors, *Discontinuous Galerkin methods. Theory, computation and applications*, Lecture Notes in Computational Science and Engineering, 11. Springer.

Cockburn, B., Qian, J., Reitich, F., and Wang, J. (2005). An accurate spectral/discontinuous finite-element formulation of a phase-space-based level set approach to geometrical optics. *Journal of Computational Physics*, 208:175–195.

Cockburn, B. and Shu, C.-W. (1989). TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework. *Mathematics of Computation*, 52:411–435.

Cockburn, B. and Shu, C.-W. (1998). The Runge–Kutta discontinuous Galerkin method for conservation laws V. *Journal of Computational Physics*, 141(2):199–224.

Cockburn, B. and Shu, C.-W. (2001). Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3):173–261.

Courant, R. and Hilbert, D. (1966). *Methods of Mathematical Physics*. John Wiley, New York.

Crandall, M. G. and Lions, P. L. (1983). Viscosity solutions of Hamilton–Jacobi equations. *Transactions of the American mathematical society*, 277(1):1–42.

Crandall, M. G. and Lions, P. L. (1984). Two approximations of solutions of Hamilton–Jacobi equations. *Mathematics of computation*, 43(167):1–19.

Crase, E., Pica, A., Noble, M., McDonald, J., and Tarantola, A. (1990). Robust elastic non-linear waveform inversion: application to real data. *Geophysics*, 55:527–538.

Cristiani, E. (2009). A fast marching method for Hamilton-Jacobi equations modeling monotone front propagations. *Journal of Scientific Computing*, 39:189–205.

de Bruin, C. G. M., Wapenaar, C. P. A., and Berkhout, A. J. (1990). Angle-dependent reflectivity by means of prestack migration. *Geophysics*, 55(9):1223–1234.

Dell, S. and Gajewski, D. (2011). Common-reflection-surface-based workflow for diffraction imaging. *Geophysics*, 76(5):S187–S195.

Dellinger, J. and Symes, W. (1997). Anisotropic finite-difference traveltimes using a Hamilton–Jacobi solver. *SEG Technical Program Expanded Abstracts*, 16(1):1786–1789.

Dey-Barsukov, S., Dürrast, H., Rabbel, W., Siegesmund, S., and Wende, S. (2000). Aligned fractures in carbonate rocks: laboratory and in situ measurements of seismic anisotropy. *International Journal of Earth Sciences*, 88(4):829–839.

Dijkstra, E. W. (1959). A note on two problems in connection with graphs. *Numerische Mathematik*, 1:269–271.

Duveneck, E. (2004). Velocity model estimation with data-derived wavefront attributes. *Geophysics*, 69(1):265–274.

Dyer, B. and Worthington, M. (1988). Seismic reflection tomography: a case study. *First Break*, 6(11):354–366.

Dziewonski, A. and Romanowicz, B. (2015). Deep earth seismology: An introduction and overview. *Treatise on Geophysics (Second Edition)*, 1:1–28.

El-Mageed, M. A., Kim, S., and Symes, W. W. (1997). 3-D Kirchhoff migration using finite difference traveltimes and amplitudes. Technical report, The Rice Inversion Project, Rice University.

Engquist, B. and Runborg, O. (2003). Computational high frequency wave propagation. *Acta Numerica*, 12:181–266.

Farra, V. (1989). Ray perturbation theory for heterogeneous hexagonal anisotropic media. *Geophysical Journal International*, 99:723–737.

Farra, V. and Madariaga, R. (1987). Seismic waveform modeling in heterogeneous media by ray perturbation theory. *Journal of Geophysical Research*, 92:2697–2712.

Farra, V. and Madariaga, R. (1988). Non-linear reflection tomography. *Geophysical Journal*, 95:135–147.

Fatemi, E., Engquist, B., and Osher, S. (1995). Numerical solution of the high frequency asymptotic expansion for the scalar wave equation. *Journal of Computational Physics*, 120:145–155.

Faye, J. P. and Jeannot, J. P. (1986). Prestack migration velocities from focusing depth analysis. In *Expanded Abstracts*, pages 438–440. Soc. Expl. Geophys.

Fichtner, A., Bunge, H. P., and Igel, H. (2006). The adjoint method in seismology: I - Theory. *Physics of the Earth and Planetary Interiors*, 157(1-2):86–104.

Fomel, S., Luo, S., and Zhao, H.-K. (2009). Fast sweeping method for the factored eikonal equation. *Journal of Computational Physics*, 228:6440–6455.

Fomel, S. and Sethian, J. A. (2012). Fast phase-space computation of multiple arrivals. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 99, pages 7329–7334.

Forgues, E. (1996). *Inversion linearisée multi-paramètres via la théorie des rais*. PhD thesis, Institut Français du Pétrole - University Paris VII.

Fornberg, B. (1996). *A Practical Guide to Pseudospectral Methods*. Cambridge Monographs on Applied and Computational Mathematics (No. 1).

Fung, Y. C. (1977). *A first course in continuum mechanics*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Gajewski, D. and Pšenčík, I. (1990). Vertical seismic profile synthetics by dynamic ray tracing in laterally varying layered anisotropic structures. *Journal of Geophysical Research: Solid Earth*, 95(B7):11301–11315.

Geoltrain, J. and Brac, S. (1993). Can we image complex structures with first-arrival traveltime? *Geophysics*, 58:564–575.

Gray, S. H. and May, W. P. (1994). Kirchhoff migration using eikonal equation traveltimes. *Geophysics*, 54:810–817.

Grechka, V., Vasconcelos, I., and Kachanov, M. (2006). The influence of crack shape on the effective elasticity of fractured rocks. *Geophysics*, 71(5):D153–D160.

Gremaud, P. A. and Kuster, C. M. (2006). Computational study of fast methods for the Eikonal equation. *SIAM Journal on Scientific Computing*, 27(6):1803–1816.

Gubbins, D. (1996). A formalism for the inversion of geomagnetic data for core motions with diffusion. *Physics of the Earth and Planetary Interiors*, 98(3–4):193–206.

Guitton, A. and Alkhalifah, T. (2013). An introduction to this special section: Full-waveform inversion and the way forward. *The Leading Edge*, September:1026–1028.

Gutenberg, B. (1914). Über erdbenwellen viia. beobachtungen an registrierungen von fernbeben in göttingen und folgerungen über die konstitution des erdkörpers. *Nachrichten von der Könglichen Gesellschaft der Wissenschaften zu Göttinge, Mathematisch-Physikalische Klasse*, pages 125–176.

Han, S., Zhang, W., and Zhang, J. (2017). Calculating qP-wave travel times in 2D TTI media by high-order fast sweeping methods with a numerical quartic equation solver. *Geophysical Journal International*, 210(3):1560–1569.

Hauser, J., Sambridge, M., and Rawlinson, N. (2006). Phase space methods for multi-arrival wavefronts. *Exploration Geophysics*, 37:331–339.

Hesthaven, J. S. and Warburton, T. (2008). *Nodal Discontinuous Galerkin Method. Algorithms, Analysis, and Application*. Springer, New York.

Hole, D. and Zelt, B. (1995). 3-D finite difference reflection traveltimes. *Geophysical Journal International*, 121:427–434.

Hole, J. A. (1992). Nonlinear high-resolution three-dimensional seismic travel time tomography. *Journal of Geophysical Research*, 97:6553–6562.

Hou, J. and Symes, W. W. (2015). An approximate inverse to the extended Born modeling operator. *Geophysics*, 80(6):R331–R349.

Hou, J. and Symes, W. W. (2016). Accelerating extended least-squares migration with weighted conjugate gradient iteration. *Geophysics*, 81:S165–S179.

Hou, J. and Symes, W. W. (2017). An alternative formula for approximate extended born inversion. *Geophysics*, 82:S1–S8.

Hou, J. and Symes, W. W. (2018). Inversion velocity analysis in the subsurface-offset domain. *Geophysics*, 83:R189–R200.

Hu, C. and Shu, C.-W. (1999). A discontinuous galerkin finite element method for Hamilton–Jacobi equations. *SIAM Journal on Scientific Computing*, 21:666–690.

Hu, J., Cao, J., Wang, H., Liu, S., and Wang, X. (2017). 3D traveltime computation for quasi-P-wave in orthorhombic media using dynamic programming. *Geophysics*, 83(1):C27.

Hubral, P. (1983). Computing true amplitude reflections in a laterally inhomogeneous earth. *Geophysics*, 48(8):1051–1062.

Improta, L., Zollo, A., Herrero, A., Frattini, R., Virieux, J., and DellÁversana, P. (2002). Seismic imaging of complex structures by non-linear traveltimes inversion of dense wide-angle data: application to a thrust belt. *Geophysical Journal International*, 151:264–278.

Jiang, G.-S. and Peng, D. (2000). Weighted ENO schemes for Hamilton–Jacobi equations. *SIAM Journal on Scientific Computing*, 21:2126–2143.

Jiang, G.-S. and Shu, C.-W. (1996). Efficient implementation of weighted eno schemes. *Journal of Computational Physics*, 126:202–228.

Jin, S., Madariaga, R., Virieux, J., and Lambaré, G. (1992). Two-dimensional asymptotic iterative elastic inversion. *Geophysical Journal International*, 108:575–588.

Julian, B. R. and Gubbins, D. (1977). Three-dimensional seismic ray tracing. *Journal of Geophysics*, 43:95–113.

Kachanov, M. (1980). Continuum model of medium with cracks. *Journal of the Engineering Mechanics Division*, 106(5):1039–1051.

Kao, C. Y., Osher, S., and Qian, J. (2004). Lax-friedrichs sweeping schemes for static hamilton-jacobi equations. *Journal of Computational Physics*, 196:367–391.

Kao, C.-Y., Osher, S., and Tsai, Y.-H. R. (2005). Fast sweeping methods for static Hamilton–Jacobi equations. *SIAM Journal on Numerical Analysis*, 42(6):2612–2632.

Käser, M. and Dumbser, M. (2006). An Arbitrary High Order Discontinuous Galerkin Method for Elastic Waves on Unstructured Meshes I: The Two-Dimensional Isotropic Case with External Source Terms. *Geophysical Journal International*, 166:855–877.

Keller, J. B. (1962). A geometrical theory of diffraction. *Journal of the Optical Society of America*, 52:116–130.

Kim, S. (1999). On Eikonal solvers for anisotropic traveltimes. *SEG Technical Program Expanded Abstracts 1999*, pages 1875–1878.

Kim, S. and Cook, R. (1999). 3d traveltime computation using second order eno scheme. *Geophysics*, 64:1867–1876.

Kimmel, R. and Sethian, J. A. (1998). Computing geodesic paths on manifolds. In *Proceedings of the National Academy of Sciences of the United States of America*, pages 8431–8435.

Kimmel, R. and Sethian, J. A. (2001). Optimal algorithm for shape from shading and path planning. *Journal of Mathematical Imaging and Vision*, 14(3):237–244.

Klein, F. W. (1978). *Hypocenter location program HYPOINVERSE: Part I. Users guide to Versions 1, 2, 3, and 4.* U.S Geological Survey Open-File Report 78–694.

Klem-Musatov, K. D. (1995). *Theory of Seismic diffraction*. Society of Exploration Geophysicists, Tulsa.

Komatitsch, D. and Vilotte, J. P. (1998). The spectral element method: an efficient tool to simulate the seismic response of 2D and 3D geological structures. *Bulletin of the Seismological Society of America*, 88:368–392.

Kouyoumjian, R. G. and Pathak, P. H. (1974). A uniform geometrical theory of diffraction for an edge in a perfectly conducting surface. *Proceedings of the IEEE*, 62(11):1448–1461.

Kravtsov, Y. A. (1964). On a modification of the method of geometrical optics. *Radiofizika*, 7:664–673.

LaFehr, T. R. (1983). Rock density from borehole gravity surveys. *Geophysics*, 48(3).

Lailly, P. (1983). The seismic inverse problem as a sequence of before stack migrations. In Bednar, R. and Weglein, editors, *Conference on Inverse Scattering, Theory and application, Society for Industrial and Applied Mathematics, Philadelphia*, pages 206–220.

Lambaré, G. (2008). Stereotomography. *Geophysics*, 73(5):VE25–VE34.

Lambaré, G., Lucio, P. S., and Hanyga, A. (1996). Two-dimensional multivalued traveltime and amplitude maps by uniform sampling of ray field. *Geophysical Journal International*, 125:584–598.

Lambaré, G., Operto, S., Podvin, P., Thierry, P., and Noble, M. (2003). 3-D ray+Born migration/inversion - part 1: theory. *Geophysics*, 68:1348–1356.

Lambaré, G., Virieux, J., Madariaga, R., and Jin, S. (1992). Iterative asymptotic inversion in the acoustic approximation. *Geophysics*, 57:1138–1154.

Lan, H. and Chen, L. (2017). An upwind fast sweeping scheme for calculating seismic wave first-arrival travel times for models with an irregular free surface. *Geophysical Prospecting*, 66(2):327–341.

Lan, H. and Zhang, Z. (2013). Topography-dependent eikonal equation and its solver for calculating first-arrival traveltimes with an irregular surface. *Geophysical Journal International*, 193(2):1010–1026.

Lay, T. and Wallace, T. C. (1995). *Modern global seismology*. Academic Press.

Le Bouteiller, P., Benjemaa, M., Chauris, H., Métivier, L., Tavakoli, B., Noble, M., and Virieux, J. (2017). Discontinuous Galerkin method for TTI Eikonal equation. In *Proceedings of the* 79^{th} *EAGE Conference & Exhibition*. EAGE.

Le Bouteiller, P., Benjemaa, M., Métivier, L., and Virieux, J. (2018a). An accurate discontinuous Galerkin method for solving point–source Eikonal equation in 2-D heterogeneous anisotropic media. *Geophysical Journal International*, 212(3):1498–1522.

Le Bouteiller, P., Benjemaa, M., Métivier, L., and Virieux, J. (2018b). A discontinuous Galerkin fast-sweeping Eikonal solver for fast and accurate traveltime computation in 3D tilted anisotropic media. *Geophysics*, in press.

Le Meur, H. (1994). *Tomographie tridimensionnelle à partir des temps des premières arrivées des ondes P et S*. PhD thesis, Université Paris VII.

Le Meur, H., Virieux, J., and Podvin, P. (1997). Seismic tomography of the gulf of Corinth: a comparison of methods. *Annali Di Geofisica*, XL(1):1–24.

Lee, W. H. and Lahr, J. C. (1975). *HYPO71 (revised); a computer program for determining hypocenter, magnitude, and first motion pattern of local earthquakes*. U.S Geological Survey Open-File Report 75–311.

Legchenko, A. (2013). *SNMR Imaging for Groundwater*, chapter 1, pages 1–14. Wiley-Blackwell.

Lehmann, I. (1936). P'. *Publications du Bureau Central Séismologique International*, A14:87–115.

Lelièvre, P. G., Farquharson, C. G., and Hurich, C. A. (2011). Inversion of first-arrival seismic traveltimes without rays, implemented on unstructured grids. *Geophysical Journal International*, 185:749–763.

Leung, S. and Qian, J. (2006). An adjoint state method for three-dimensional transmission traveltime tomography using first-arrivals. *Communications in Mathematical Sciences*, 4(1):249–266.

LeVeque, R. J. (2002). *Finite volume methods for hyperbolic problems*. Cambridge University Press, New-York, USA.

LeVeque, R. J. (2007). *Finite Difference Methods for Ordinary and Partial Differential Equations, Steady State and Time Dependent Problems*. SIAM, Philadelphia, USA.

Levin, F. K. (1979). Reply to K. Helbig by F. K. Levin. *Geophysics*, 44(5):990–990.

Li, F., Shu, C. W., Zhang, Y. T., and Zhao, H. (2008). A second order discontinuous galerkin fast sweeping method for Eikonal equations. *Journal of Computational Physics*, 227:8191–8208.

Li, P. and Lin, G. (2016). Local earthquake tomography with the inclusion of full topography and its application to kīlauea volcano, hawaii. *Journal of Volcanology and Geothermal Research*, 316:12–21.

Li, S. and Fomel, S. (2013). Kirchhoff migration using eikonal-based computation of traveltime source-derivatives. *Geophysics*, 78(4):S211–S219.

Li, S., Vladimirsky, A., and Fomel, S. (2013). First-break traveltime tomography with the double-square-root eikonal equation. *Geophysics*, 78(6):U89–U101.

Li, Y. and Chauris, H. (2018). Coupling direct inversion to common-shot image-domain velocity analysis. *Geophysics*, 83(5):R497–R514.

Li, Y., Han, W., song Chen, C., and Huang, T. (2012). Velocity model building for tilted orthorhombic depth imaging. In *SEG Technical Program Expanded Abstracts 2012*.

Lifschitz, A. and Hameiri, E. (1991). Local stability conditions in fluid dynamics. *Physics of Fluids. A: Fluid Dynamics*, 3(11):2644–2651.

Lions, P.-L. (1982). *Generalized solutions of Hamilton–Jacobi equations*. Pitman.

Liu, H., Osher, S., and Tsai, R. (2006). Multi-valued solution and level set methods in computational high frequency wave propagation. *Communications in Computational Physics*, 1(5):765–804.

Liu, X., Osher, S., and Chan, T. (1994). Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115:200–212.

Lomax, A., Virieux, J., Volant, P., and Berge, C. (2000). Probabilistic earthquake location in 3D and layered models: Introduction of a metropolis-gibbs method and comparison with linear locations. In Thurber, C. H. and Rabinowitz, N., editors, *Advances in Seismic Event Location*, volume 281, pages 101–134. Kluwer Academic Pub., Amsterdam.

Ludwig, D. (1966). Uniform asymptotic expansions at a caustic. *Communications on Pure and Applied Mathematics*, 19:215–250.

Luo, S. and Qian, J. (2011). Factored singularities and high-order Lax-Friedrichs sweeping schemes for point-source traveltimes and amplitudes. *Journal of Computational Physics*, 230:4742–4755.

Luo, S. and Qian, J. (2012). Fast sweeping method for factored anisotropic eikonal equations: multiplicative and additive factors. *Journal of Scientific Computing*, 52:360–382.

Luo, S., Qian, J., and Burridge, R. (2014a). Fast huygens sweeping methods for Helmholtz equations in inhomogeneous media in the high frequency regime. *Journal of Computational Physics*, 270:378–401.

Luo, S., Qian, J., and Burridge, R. (2014b). High-order factorization based high-order hybrid fast sweeping methods for point source eikonal equations. *SIAM Journal on Numerical Analysis*, 52:23–44.

Luo, S., Qian, J., and Zhao, H.-K. (2012). Higher-order schemes for 3D first-arrival traveltimes and amplitudes. *Geophysics*, 77:T47–T56.

Luo, S. and Zhao, H. (2016). Convergence analysis of the fast sweeping method for static convex Hamilton–Jacobi equations. *Research in the Mathematical Sciences*, 3(1):35.

Marfurt, K. (1984). Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations. *Geophysics*, 49:533–549.

Mendes, M. (2000). Green's function interpolations for prestack imaging. *Geophysical Prospecting*, 48:49–62.

Menke, W. (1984). *Geophysical Data Analysis: Discrete Inverse Theory*. Academic Press, Inc., Orlando, USA.

Mensch, T. and Farra, V. (2002). *P-wave Tomography in Inhomogeneous Orthorhombic Media*, pages 1855–1879. Birkhäuser Basel.

Mensch, T., Farra, V., and Singh, S. C. (1998). 3D seismic traveltime tomography in orthorhombic media. In *Expanded Abstracts*, volume II, pages 1863–1866. Society of Exploration Geophysics.

Métivier, L., Brossier, R., and Virieux, J. (2015). Combining asymptotic linearized inversion and full waveform inversion. *Geophysical Journal International*, 201(3):1682–1703.

Miller, D., Oristaglio, M., and Beylkin, G. (1987). A new slant on seismic imaging: Migration and integral geometry. *Geophysics*, 52(7):943–964.

Mirebeau, J.-M. and Portegies, J. (2018). Hamiltonian fast marching: A numerical solver for anisotropic and non-holonomic eikonal pdes. Preprint.

Moczo, P., Kristek, J., and Halada, L. (2004). *The Finite-Difference Method for Seismologists An Introduction*. Comenius University Bratislava.

Mohorovičić, A. (1909). Das beben vom 8. x. 1909. *Jb. Met. Obs. Zagreb (Agram)*, 9:1–63.

Moser, T. J. (1991). Shortest path calculation of seismic rays. *Geophysics*, 56(1):59–67.

Musgrave, M. (1970). *Crystal acoustics; introduction to the study of elastic waves and vibrations in crystals*. Holden-Day.

Nelson, G. D. and Vidale, J. (1990). Earthquake locations by 3-D finite-difference travel times. *Bulletin of the Seismological Society of America*, 80(2):395–410.

Noble, M., Gesret, A., and Belayouni, N. (2014). Accurate 3-d finite difference computation of travel time in strongly hetrogeneous media. *Geophysical Journal International*, 199:1572–1585.

Noble, M., Thierry, P., Taillandier, C., and Calandra, H. (2010). High-performance 3D first-arrival traveltime tomography. *The Leading Edge*, 29(1):86–93.

Nolet, G., Allen, R., and Zhao, D. (2007). Mantle plume tomography. *Chemical Geology*, 241(3):248–263.

Oldham, R. (1906). The constitution of the earth. *Quarterly Journal of the Geological Society of London*, 62:456–475.

Operto, S., Lambaré, G., Podvin, P., and Thierry, P. (2003). 3-D ray-Born migration/inversion. part 2: application to the SEG/EAGE overthrust experiment. *Geophysics*, 68(4):1357–1370.

Osher, S. (1993). A level set formulation for the solution of the dirichlet problem for Hamilton–Jacobi equations. *SIAM Journal on Mathematical Analysis*, 24:1145–1152.

Osher, S., Cheng, L.-T., Kang, M., Shim, H., and Tsai, Y.-H. (2002). Geometric optics in a phase-space-based level set and eulerian framework. *Journal of Computational Physics*, 179:622–648.

Osher, S. and Sethian, J. (1988). Fronts propagating with curvature-dependent speed - algorithms based on Hamilton–Jacobi formulations. *Journal of Computational Physics*, 79:12–49.

Osypov, K. (2000). Robust refraction tomography. In *70^{th} Annual International SEG Meeting, Expanded Abstracts*, pages 2032–2035.

Payton, R. (1983). *Elastic wave propagation in transversely isotropic media*. Springer Netherlands.

Peng, C., Wang, M., Chazalnoel, N., and Gomes, A. (2018). Subsalt imaging improvement possibilities through a combination of fwi and reflection fwi. *The Leading Edge*, 37(1):52–57.

Pica, A. (1997). Fast and accurate finite-difference solutions of the 3-D eikonal equation parameterized in celerity. In *Expanded Abstracts*, pages 1774–1777. Society of Exploration Geophysics.

Pica, A., Diet, J. P., and Tarantola, A. (1990). Nonlinear inversion of seismic reflection data in laterally invariant medium. *Geophysics*, 55(3):284–292.

Piperno, S. (2005). DGTD methods using modal basis functions and symplectic local time-stepping: application to wave propagation problems. Technical Report RR-5749, INRIA.

Plessix, R. E. (1996). *Détermination de la vitesse pour l'interprétation de données sismiques très haute résolution à l'échelle géotechnique*. PhD thesis, Université Paris IX Dauphine.

Plessix, R. E. (2006). A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, 167(2):495–503.

Plewa, T., Linde, T., and Weirs, V. G., editors (2003). *Adaptive Mesh Refinement - Theory and Applications. Proceedings of the Chicago Workshop on Adaptive Mesh Refinement Methods, Sept. 3–5, 2003*. Springer.

Podvin, P. and Lecomte, I. (1991). Finite difference computation of traveltimes in very contrasted velocity model : a massively parallel approach and its associated tools. *Geophysical Journal International*, 105:271–284.

Popov, M. M. (1982). A new method of computation of wave fields using gaussian beams. *Wave Motion*, 4:85–95.

Postma, G. (1955). Wave propagation in a stratified medium. *Geophysics*, 20(4):780–806.

Qian, J. and Leung, S. (2004). A level set based Eulerian method for paraxial multivalued traveltimes. *Journal of Computational Physics*, 197:711–736.

Qian, J. and Symes, W. (2002a). An adaptive finite-difference method for traveltimes and amplitudes. *Geophysics*, 67:167–176.

Qian, J. and Symes, W. W. (2001). Paraxial eikonal solvers for anisotropic quasi-p travel times. *Journal of Computational Physics*, 173:256–278.

Qian, J. and Symes, W. W. (2002b). Finite-difference quasi-P traveltimes for anisotropic media. *Geophysics*, 67:147–155.

Qian, J., Symes, W. W., and Dellinger, J. A. (2001). A full-aperture anisotropic Eikonal solver for quasi-P traveltimes. *SEG Technical Program Expanded Abstracts 2001*, pages 129–132.

Qian, J., Zhang, Y.-T., and Zhao, H.-K. (2007a). A fast sweeping method for static convex Hamilton–Jacobi equations. *Journal of Scientific Computing*, 31:237–271.

Qian, J., Zhang, Y. T., and Zhao, H. K. (2007b). Fast sweeping methods for Eikonal equations on triangular meshes. *SIAM Journal on Numerical Analysis*, 45(1):83–107.

Qin, B., Allemand, T., and Lambaré, G. (2015). Full waveform inversion using preserved amplitude reverse time migration. In *SEG Technical Program Expanded Abstracts 2015*, pages 1252–1257.

Qin, F., Luo, Y., Olsen, K. B., Cai, W., and Schuster, G. T. (1992). Finite difference solution of the eikonal equation along expanding wavefronts. *Geophysics*, 57:478–487.

Qin, F. and Schuster, G. (1993). First-arrival travel-time calculation for anisotropic media. *Geophysics*, 58:1349–1358.

Ralston, J. (1982). Gaussian beams and the propagation of singularities. *Studies in partial differential equations*, 23(206).

Rawlinson, N., Fichtner, A., Sambridge, M., and Young, M. K. (2014). Seismic tomography and the assessment of uncertainty. In *Advances in Geophysics*, volume 55, pages 1–76. Elsevier.

Rawlinson, N., Hauser, J., and Sambridge, M. (2007). Seismic ray tracing and wavefront tracking in laterally heterogeneous media. *Advances in Geophysics*, 49:203–267.

Rawlinson, N., Pozgay, S., and Fishwick, S. (2010). Seismic tomography: a window into deep earth. *Physics of the Earth and Planetary Interiors*, 178(3):101–135.

Rawlinson, N. and Sambridge, M. (2003). Seismic traveltime tomography of the crust and lithosphere. *Advances in Geophysics*, 46:81–198.

Reed, W. and Hill, T. (1973). Triangular mesh methods for the neuron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory.

Regone, C., Stefani, J., Wang, P., Gerea, C., Gonzalez, G., and Oristaglio, M. (2017). Geologic model building in SEAM Phase II - Land seismic challenges. *The Leading Edge*, 36(9):738–749.

Riznichenko, Y. V. (1946). Geometrical seismics of layered media. *Trudy Inst. Theor. Geophysics*, 2.

Rouy, E. and Tourin, A. (1992). A viscosity solution approach to shape—from—shading. *SIAM Journal on Numerical Analysis*, 29(3):867–884.

Runborg, O. (2007). Mathematical models and numerical methods for high frequency waves. *Communications in Computational Physics*, 2(5):827–880.

Sambolian, S., Operto, S., Ribodetti, A., Tavakoli F., B., and Virieux, J. (2018). Parsimonious slope tomography based on eikonal solvers and the adjoint-state method. In *Expanded Abstracts, 80^{th} Annual EAGE Meeting (Copenhagen)*.

Sava, P. and Fomel, S. (2006). Time-shift imaging condition in seismic migration. *Geophysics*, 71(6):S209–S217.

Schiesser, W. E. (1991). *The numerical method of lines : integration of partial differential equations*. Academic Press.

Schijns, H., Schmitt, D. R., Heikkinen, P. J., and Kukkonen, I. T. (2012). Seismic anisotropy in the crystalline upper crust: observations and modelling from the outokumpu scientific borehole, Finland. *Geophysical Journal International*, 189(1):541–553.

Schwarz, B. and Gajewski, D. (2017). A generalized view on normal moveout. *Geophysics*, 82(5):V335–V349.

Schwarz, B., Vanelle, C., Gajewski, D., and Kashtan, B. (2014). Curvatures and inhomogeneities: An improved common-reflection-surface approach. *Geophysics*, 79(5):S231–S240.

Sethian, J. A. (1996). A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences of the United States of America*, 93:1591–1595.

Sethian, J. A. (1999). Fast marching methods. *SIAM Review*, 41(2):199–235.

Shah, H. (2007). The 2007 bp anisotropic velocity-analysis benchmark. In *Expanded Abstracts*. EAGE workshop.

Shu, C.-W. and Osher, S. (1988). Efficent implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77:439–471.

Shu, C.-W. and Osher, S. (1989). Efficent implementation of essentially non-oscillatory shock-capturing schemes, ii. *Journal of Computational Physics*, 83:32–78.

Slawinski, M. A. (2003). *Seismic Waves and Rays in Elastic Media*. Elsevier Science.

Stein, S. and Wysession, M. (2003). *An Introduction to Seismology, Earthquakes and Earth Structure*. Blackwell Publishing.

Sun, J., Sun, Z., and Han, F. (2011). A finite difference scheme for solving the eikonal equation including surface topography. *Geophysics*, 76(4):T53–T63.

Sun, Y. (1992). Computation of 2D multiple arrival traveltime fields by an interpolative shooting method. In *Expanded Abstracts*, pages 1320–1323. Society of Exploration Geophysics.

Sykes, L. R. (1967). Mechanism of earthquakes and nature of faulting on the mid-oceanic ridges. *Journal of Geophysical Research*, 72(8).

Symes, W. W. (2008). Migration velocity analysis and waveform inversion. *Geophysical Prospecting*, 56:765–790.

Symes, W. W. (2009). The seismic reflection inverse problem. *Inverse Problems*, 25(12):123008.

Symes, W. W. and Qian, J. (2003). A slowness matching eulerian method for multivalued solutions of eikonal equation. *Journal of Scientific Computing*, 19:501–525.

Taflove, A. and Hagness, S. C. (2005). *Computational Electrodynamics: The Finite-Difference Time-Domain Method*. Artech House, 3rd edition.

Taillandier, C., Noble, M., Chauris, H., and Calandra, H. (2009). First-arrival travel time tomography based on the adjoint state method. *Geophysics*, 74(6):WCB1–WCB10.

Tarantola, A. (1984). Inversion of seismic reflection data in the acoustic approximation. *Geophysics*, 49(8):1259–1266.

Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. Society for Industrial and Applied Mathematics, Philadelphia.

Tarrass, I., Giraud, L., and Thore, P. (2011). New curvilinear scheme for elastic wave propagation in presence of curved topography. *Geophysical Prospecting*, 59(5):889–906.

Tavakoli F., B., Operto, S., Ribodetti, A., and Virieux, J. (2018). Matrix-free anisotropic slope tomography: theory and application. *Geophysics*, Accepted for publication, Posted online on 18 Sep 2018.

Tavakoli F., B., Operto, S., Ribodetti, A., and Virieux, J. (2017). Slope tomography based on eikonal solvers and the adjoint-state method. *Geophysical Journal International*, 209(3):1629–1647.

Tavakoli F., B., Ribodetti, A., Virieux, J., and Operto, S. (2015). An iterative factored eikonal solver for TTI media. In *SEG technical program expanded abstracts 2015*, volume 687, pages 3576–3581.

ten Kroode, F. (2012). A wave-equation-based kirchhoff operator. *Inverse Problems*, 28(11):115013.

Thanoon, D., Bachrach, R., Zdraveva, O., and Chen, S. (2015). Tilted orthorhombic model building with geomechanics: Theory and observations. In *Expanded Abstracts, 77th Annual EAGE Meeting (Vienna)*.

Thomsen, L. A. (1986). Weak elastic anisotropy. *Geophysics*, 51:1954–1966.

Thornburgh, H. R. (1930). Wave-front diagrams in seismic interpretation. *AAPG Bulletin*, 14(2):185–200.

Thurber, C. and Ritsema, J. (2007). Theory and observations - seismic tomography and inverse methods. In Romanowicz, B. and Diewonski, A., editors, *Treatise of Geophysics, volume 1: Seismology and structure of the Earth, Treatise of Geophysics*. Elsevier.

Trinh, P. T., Brossier, R., Métivier, L., Tavard, L., and Virieux, J. (2017). Efficient 3D elastic FWI using a spectral-element method. In *87th SEG Conference and Exhibition 2017, Houston*, pages 1533–1538.

Trinh, P. T., Brossier, R., Métivier, L., Tavard, L., and Virieux, J. (2018). Efficient 3D time-domain elastic and viscoelastic Full Waveform Inversion using a spectral-element method on flexible Cartesian-based mesh. *Geophysics*.

Tsai, Y.-H. R., Chen, L.-T., Osher, S., and Zhao, H.-K. (2003). Fast sweeping algorithms for a class of Hamilton–Jacobi equations. *SIAM Journal on Numerical Analysis*, 41(2):673–694.

Tsitsiklis, J. N. (1995). Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, 40(9):1528–1538.

Tsvankin, I. (1997). Anisotropic parameters and P-wave velocity for orthorhombic media. *Geophysics*, 62:1292–1309.

Tsvankin, I., Gaiser, J., Grechka, V., van der Baan, M., and Thomsen, L. (2010). Seismic anisotropy in exploration and reservoir characterization: An overview. *Geophysics*, 75(5):75A15–75A29.

van Trier, J. and Symes, W. W. (1991). Upwind finite-difference calculation of traveltimes. *Geophysics*, 56(6):812–821.

Vanelle, C. and Gajewski, D. (2002). Second-order interpolation of traveltimes. *Geophysical Prospecting*, 50:73–83.

Vidale, D. (1988a). Finite-difference calculation of travel time. *Bulletin of the Seismological Society of America*, 78:2062–2076.

Vidale, D. (1988b). Finite-difference calculation of travel time. *Bulletin of the Seismological Society of America*, 78:2062–2076.

Vidale, D. (1990). Finite-difference calculation of traveltimes in three dimensions. *Geophysics*, 55:521–526.

Vinje, V., Iversen, E., Astebøl, K., and Gjøystdal, H. (1996a). Estimation of multivalued arrivals in 3D models using wavefront construction- Part II: Tracing and interpolation. *Geophysical Prospecting*, 44:843–858.

Vinje, V., Iversen, E., Astebøl, K., and Gjøystdal, H. (1996b). Estimation of multivalued arrivals in 3D models using wavefront construction-part I. *Geophysical Prospecting*, pages 819–842.

Vinje, V., Iversen, E., and Gjøystdal, H. (1993). Traveltime and amplitude estimation using wavefront construction. *Geophysics*, pages 1157–1166.

Virieux, J. (1986a). P-SV wave propagation in heterogeneous media: Velocity-stress finite difference method. *Geophysics*, 51:889–901.

Virieux, J. (1986b). P-SV wave propagation in heterogeneous media: Velocity-stress finite difference method. *Geophysics*, 51:889–901.

Virieux, J. (1996). Seismic ray tracing. In Boschi, E., Ekström, G., and Morelli, A., editors, *Seismic modelling of Earth structure*, pages 223–304. Editrice Compositori, Bologna.

Virieux, J., Asnaashari, A., Brossier, R., Métivier, L., Ribodetti, A., and Zhou, W. (2017). An introduction to Full Waveform Inversion. In Grechka, V. and Wapenaar, K., editors, *Encyclopedia of Exploration Geophysics*, pages R1–1–R1–40. Society of Exploration Geophysics.

Virieux, J., Calandra, H., and Plessix, R. E. (2011). A review of the spectral, pseudo-spectral, finite-difference and finite-element modelling techniques for geophysical imaging. *Geophysical Prospecting*, 59:794–813.

Virieux, J. and Farra, V. (1991). Ray tracing in 3D complex isotropic media: an analysis of the problem. *Geophysics*, 16:2057–2069.

Virieux, J. and Lambaré, G. (2015). Theory and observations: Body waves, ray methods, and finite-frequency effects. *Treatise on Geophysics (Second Edition)*, 1:169–200.

Virieux, J. and Operto, S. (2009). An overview of full waveform inversion in exploration geophysics. *Geophysics*, 74(6):WCC1–WCC26.

von Westenholz, C. (1977). On the geometric structure underlying the eikonal equation. *Foundations of Physics*, 7(7–8):529–547.

Waheed, U. and Alkhalifah, T. (2017). A fast sweeping algorithm for accurate solution of the tilted transversely isotropic eikonal equation using factorization. *Geophysics*, 82(6).

Waheed, U., Alkhalifah, T., and Wang, H. (2015a). Efficient traveltime solutions of the acoustic TI eikonal equation. *Journal of Computational Physics*, 282(Supplement C):62–76.

Waheed, U. B., Flagg, G., and Yarman, C. E. (2016). First-arrival traveltime tomography for anisotropic media using the adjoint-state method. *Geophysics*, 81(4):R147–R155.

Waheed, U. B., Yarman, C. E., and Flagg, G. (2015b). An iterative, fast-sweeping-based eikonal solver for 3D tilted anisotropic media. *Geophysics*, 80:C49–C58.

Wilson, J. T. (1965). A new class of faults and their bearing on continental drift. *Nature*, 207:343–347.

Woodhouse, J. and Dziewonski, A. (1984). Mapping the upper mantle: Three dimensional modelling of earth structure by inversion of seismic waveforms. *Journal of Geophysical Research*, 89:5953–5986.

Yilmaz, O. (2001). *Seismic data analysis*. Society of Exploration Geophysicists: processing, inversion and interpretation of seismic data.

Zelt, C. and Barton, P. J. (1998). Three-dimensional seismic refraction tomography: a comparison of two methods applied to data from the Faeroe basin. *Journal of Geophysical Research*, 103(B4):7187–7210.

Zhang, L., Rector III, J. W., and Hoversten, G. M. (2005a). Eikonal solver in the celerity domain. *Geophysical Journal International*, 162:1–8.

Zhang, Y., Zhang, G., and Bleistein, N. (2005b). Theory of true-amplitude one-way wave equations and true-amplitude common-shot migration. *Geophysics*, 70(4):E1–E10.

Zhang, Y., Zhao, H., and Chen, S. (2005c). Fixed–point iterative sweeping methods for static Hamilton–Jacobi equations. *Methods and Applications of Analysis*, 13(3):299–320.

Zhang, Y.-T., Chen, S., Li, F., Zhao, H., and Shu, C.-W. (2011). Uniformly accurate discontinuous Galerkin fast sweeping methods for Eikonal equations. *SIAM Journal on Scientific Computing*, 33(4):1873 – 1896.

Zhang, Y.-T., Zhao, H.-K., and Qian, J. (2006). High order fast sweeping methods for static Hamilton–Jacobi equations. *Journal of Scientific Computing*, 29:25–56.

Zhao, H. (2005). A fast sweeping method for eikonal equations. *Mathematics of computation*, 74:603–627.

Zhu, T. and Chun, K.-Y. (1994). Understanding finite-frequency wave phenomena: phase-ray formulation and inhomogeneity scattering. *Geophysical Journal International*, 119:78–90.

Zhu, X., Sixta, D. P., and Angstman, B. G. (1992). Tomostatics: Turning-ray tomography + static corrections. *The Leading Edge*, 11(12):15–23.

Zienkiewicz, O. C. and Morgan, K. (2006). *Finite elements and approximation*. Courier Corporation.