Multi Arrival Wavefront Tracking and Seismic Imaging

Jürg Hauser

October 2007

A thesis submitted for the degree of Doctor of Philosophy
of The Australian National University

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university or institution. To the best of my knowledge this thesis contains no material previously published or written by another person except where due reference is made in the text.

Jürg Hauser
October, 2007

Acknowledgements

This work would not have succeeded without the assistance and support of those people around me.

First and foremost I would like to thank Dr. Malcolm Sambridge and Dr. Nick Rawlinson. This thesis greatly benefited from their knowledge, support and encouragement. Their guidance and unfailing high standards significantly improved the end product.

The feedback provided by Prof. Brian Kennett on the occasion of my midterm review was invaluable.

I wish to thank Prof. Douglas Wiens for providing a copy of the velocity model for the seismic structure beneath the Tonga arc and Lau backarc basin.

I also would like to acknowledge the financial support provided by the Australian Research Council, and the Research School of Earth Sciences at the Australian National University.

Abstract

Multi Arrival Wavefront Tracking and Seismic Imaging

Multi valued travel times have traditionally not been used in seismic imaging, with only a handful of notable exceptions in the field of exploration geophysics. For studies on local and regional scales (e.g. local earthquake/teleseismic tomography), the focus has largely been on first arrivals. Numerous ray and grid based schemes have been developed for predicting this class of data. However, later arrivals often contribute to the length and shape of a recorded wavetrain, particularly in regions of complex geology. These arrivals are likely to contain additional information about seismic structure, as their two point path differs from that of the first arrival; in particular, they are more amenable to sampling regions of lower velocity.

Recently, several grid based schemes have been proposed for solving the multi valued travel time problem. Here a level set based scheme is investigated for its potential to accurately and robustly compute travel times in a seismological context. Although promising, it is shown that it, and other grid based multi arrival solvers, currently require significant computational resources for crustal scale problems, and further development is required before practical application becomes feasible. The main focus of this work is therefore on an alternative approach, sometimes referred to as wavefront construction. The wavefront construction principle is used as the basis of a new scheme for computing multi valued travel times that arise from smooth variations in both velocity structure and interface geometry. The idea is to represent the wavefront as a set of points, and use local ray tracing and interpolation to advance the wavefront in a series of time steps. The wavefront tracking is performed in reduced phase space, which significantly enhances the method's ability to correctly resolve complex features such as triplications. The scheme is robust in the presence of strong velocity heterogeneity and interface curvature, with phases comprising multiple reflections, refractions and triplications successfully tracked.

It is shown that using later arrivals in seismic tomography can result in better images of the subsurface. They not only improve the quality of velocity models but also of interface structure. Identifying later arrivals in observations, however, is a problem in itself. The wavefront tracking scheme is therefore extended to compute relative amplitudes and ray based seismograms which may help to facilitate this process.

In three dimensions, the wavefront becomes a surface and can be described using a mosaic of triangles. Evolving a complex surface with a given accuracy is a well known problem in the field of computer graphics. Schemes developed in this field for surface refinement and simplification are applied for the first time to a propagating wavefront in order to maintain a fixed density of nodes. This forms the basis of the scheme, proposed in this work, for tracking wavefronts in the presence of complex three dimensional velocity heterogeneity. Application of the new scheme to models containing strong velocity contrasts, including the SEG/EAGE salt dome model, demonstrate it to be robust and efficient for practical application.

Contents

Appendices

Chapter 1

Introduction

1.1 Motivation

Both continuous and discontinuous variations in wave speed can cause seismic energy to travel to a receiver along more than one path, a phenomenon commonly referred to as multipathing. This is illustrated in figure 1.1 where a wavefront triplicates due to the presence of a low velocity anomaly, resulting in the detection of three separate arrivals at the receiver. The shape of the self-intersecting wavefront at time $t + \Delta t$ resembles what is often described as a swallowtail. The first arrival path avoids the low velocity anomaly, which is subsequently sampled by the second and third arrival. Clearly, later arrivals sample different parts of the medium and therefore should carry additional structural information. However, current state of the art algorithms for tracking wavefronts or rays only provide the travel times of first arrivals (e.g. Rawlinson & Sambridge, 2004a; Buske & Kästner, 2004; de Kool et al., 2006). The development of advanced and computationally practical schemes for tracking multiple arrivals through complex two and three dimensional media would allow the prediction of a far greater proportion of the seismic wavefield, which has the potential to benefit many areas of seismology. For example, seismic imaging schemes which also exploit later portions of the recorded wavetrain could result in more detailed and accurate maps of earth structure. Figure 1.2 shows an idealised schematic plot of how multi arrival wavefront tracking compares to other common seismic wave simulation techniques with regard to representation of frequency and arrival information. While it can predict all arrivals, it is limited to the high frequency approximation of the wave equation.

One might argue that finite difference, finite element or spectral element solvers

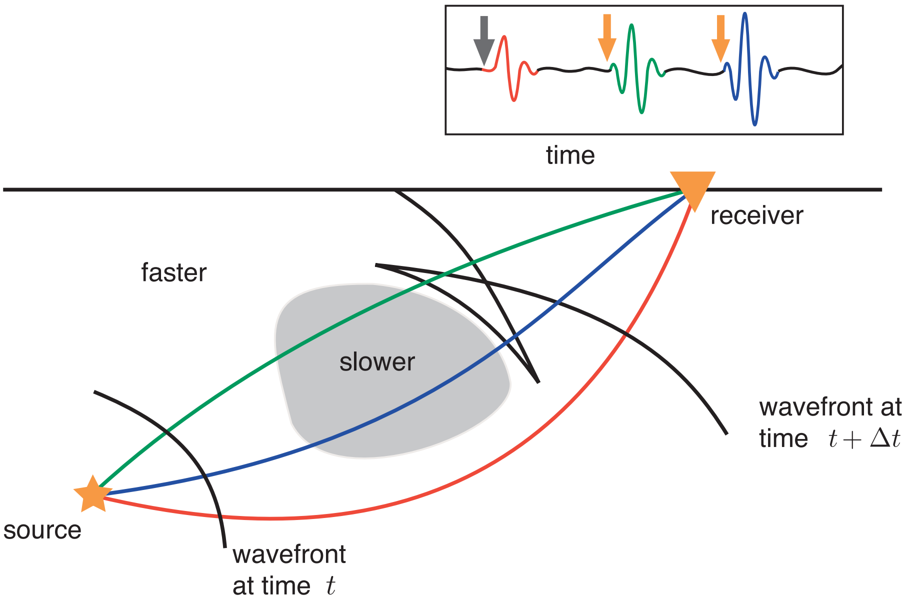


Figure 1.1: Schematic diagram showing ray paths for a medium containing a slow velocity anomaly. The wavefront triplicates and three arrivals are observed. The ray path for the first, second and third arrivals are shown in red, green and blue, respectively.

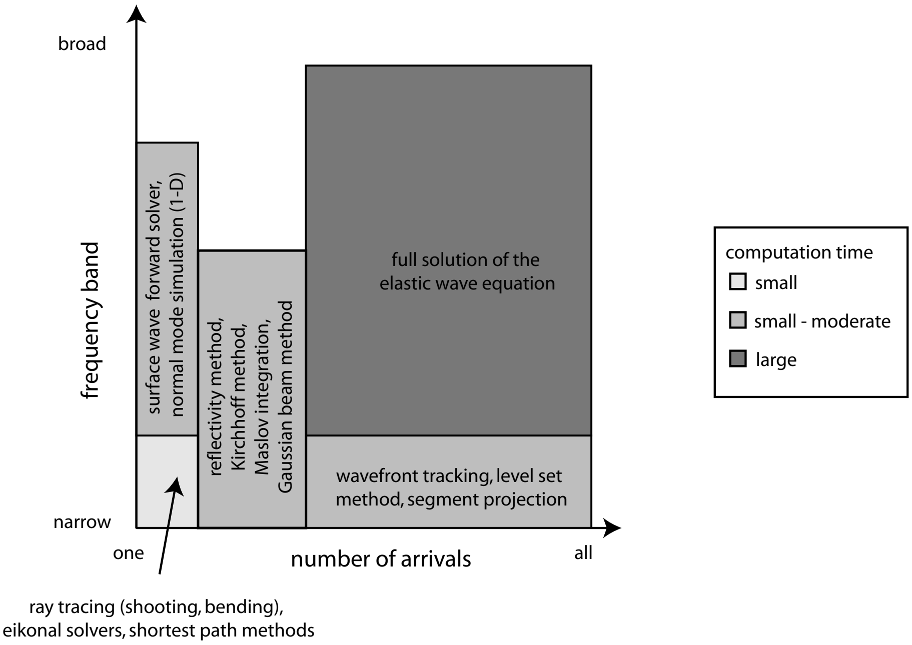

broad
narrow
frequency band
one
number of arrivals
all
surface wave forward solver, normal mode simulation (1-D)
reflectivity method, Kirchhoff method, Maslov integration, Gaussian beam method
full solution of the elastic wave equation
wavefront tracking, level set method, segment projection
ray tracing (shooting, bending), eikonal solvers, shortest path methods
computation time
□ small
▨ small - moderate
■ large

Figure 1.2: Different methods commonly used to predict information contained in observed seismograms, plotted with respect to frequency band and number of arrivals.

of the elastic wave equation (e.g. Kelly et al., 1976; Graves, 1996; Komatitsch & Tromp, 1999; Bohlen, 2002) provide synthetic seismograms which already contain later arrivals, and hence there is no need for a high frequency approach. However, numerical solution of the full wave equation is computationally expensive and significant challenges confront its use in seismic imaging. While it is true that recent developments involving adjoint methods and scattering integral methods allow path sensitivity information to be extracted (e.g. Tromp et al., 2005; Chen et al., 2007a) and used in the gradient based inversion of the synthetic waveform, many issues, including non-linearity, stability and computation time, are yet to be resolved (e.g. Tape et al., 2007; Chen et al., 2007b).

The calculation of ray travel times through a medium with a heterogeneous velocity distribution still remains the foundation of many applications that rely on the high frequency component of seismic records, such as body wave tomography, migration of reflection data and earthquake relocation (e.g. Thurber, 1983; Gray & May, 1994; Hammer et al., 1994; Steck et al., 1998). Despite many decades of technique development (e.g. Julian & Gubbins, 1977; Vidale, 1988; Sethian & Popovici, 1999), there is still no single method that can accurately, efficiently and robustly overcome the non-linearity of the two point problem, and compute all multi arrivals in complex media. The aim of this thesis is to advance the current state of the art in seismic wavefront tracking in heterogeneous media, and to investigate the potential of multi arrival information for improving various seismic applications including tomography.

1.2 Ray based methods

Traditionally, the method of choice for the computation of travel times has been ray tracing (e.g. Julian & Gubbins, 1977; Pereyra et al., 1980; Zelt & Ellis, 1988; Virieux & Farra, 1991; Červený, 2001). Ray tracing between a source and receiver can be achieved by shooting or bending rays. The shooting method relies on repeated solution of an initial value problem in order to locate two points paths. Rays are initiated at the source point with different initial directions and tracked through the medium by solving the appropriate initial value formulation of the ray tracing equation (e.g. Červený et al., 1977; Červený, 2001). This step is highly accurate and efficient, even in the presence of interfaces, and allows various seismic wave properties, like geometrical spreading, attenuation and amplitudes to be predicted. There exists a wide variety of methods for the computation of syn-

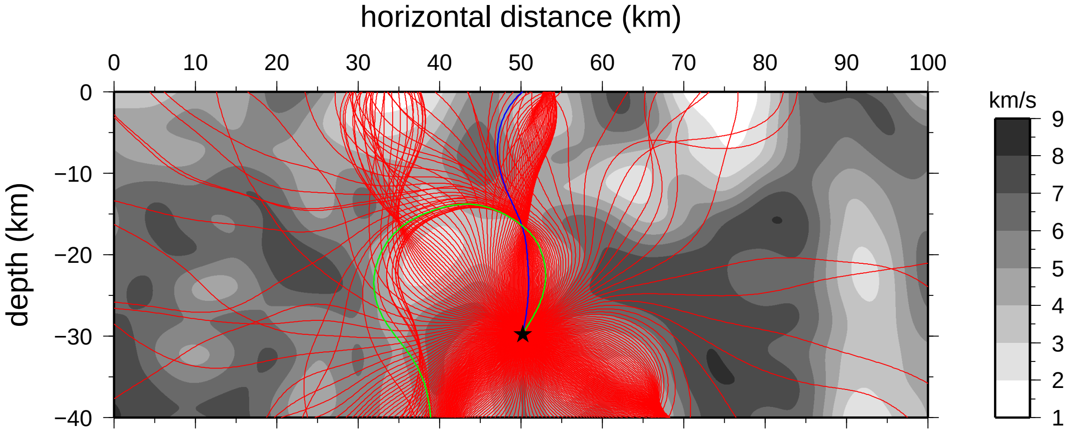


Figure 1.3: Rays generated by a uniform fan of 300 rays emitted from a source point (black star) in a smoothly varying heterogeneous velocity model. The angular distance between all adjectant paths is the same at the source. This relationship is no longer preserved as the rays are traced through the structure. Two rays with similar initial directions but very different overall geometries are marked in green and blue.

thetic seismograms based on these properties. For a layered structure one can use the reflection matrix approach of Kennett (1983) which allows the computation of a synthetic seismogram based on travel times and reflection and transmission coefficients at the interfaces. If geometrical spreading factors are computed and the number of caustics is known in advance, then the Maslov integration method can be applied (e.g. Chapman, 1985). Another approach is the so called Gaussian beam method of Červený & Pšenčík (1984), which does not require the number of caustics along a ray to be known. It is also possible to take attenuation into account by computing a dissipation factor and convolving it with the seismogram (e.g. Weber, 1988).

The real challenge of boundary value ray tracing is to determine the initial direction vector of the ray that will hit a particular receiver. This two point problem of finding a source-receiver ray path can be formulated as an inverse problem, in which the unknown is the initial direction vector of the ray, and the function to be minimised is a measure of the distance between the ray endpoint and receiver. Since the optimisation problem is non-linear, a range of iterative non-linear and fully non-linear schemes has been employed (e.g. Julian & Gubbins, 1977; Sambridge & Kennett, 1990; Virieux & Farra, 1991; Velis & Ulrych, 1996).

Figure 1.3 illustrates how the focusing and defocusing of rays in complex models makes it difficult to determine the initial direction of a ray so that it passes through a specified point in the structure. For example, no rays sample the upper right sector of the model (horizontal distance $x > 85\,\text{km}$ and depth $z > -20\,\text{km}$). There is significant defocusing of rays heading in this direction due to the fast region to the right of the source. Several rays end up propagating in a direction which differs by up to 180 ° from their initial direction. One commonly refers to these as overturning rays (see green ray in figure 1.3). Small changes in the initial ray direction also have the potential to cause significant changes in the geometry of the resulting ray path (cf. the green and blue rays in figure 1.3)

In the shooting method, the medium is normally probed with an initial ray and then information from the computed paths is used in a gradient based iterative non-linear scheme to better target the receiver (e.g. Julian & Gubbins, 1977; Sambridge & Kennett, 1990). In order to obtain an accurate initial path, some schemes shoot a broad fan of rays towards the receiver and then adjust the fan until one or more rays pass close to the receiver (e.g. Virieux & Farra, 1991; Rawlinson et al., 2001). Several shooting schemes rely on wavefront curvature information to update these initial rays until the path eventually intersects the receiver (e.g. Červený & Firbas, 1984; Sambridge & Kennett, 1990; Virieux & Farra, 1991; Bulant, 1999). Unless all possible initial directions are tried, there is no guarantee that the located path corresponds to the first or a later arrival ray path. Typically, the presence of interfaces will increase the non-linearity of the two point problem. For a complex interface, even a small perturbation of the ray intersection point can have a large impact on the direction of the outgoing ray. Two rays with small differences in their initial direction could therefore have paths which are fundamentally different.

There are relatively few examples of fully non-linear solvers in the literature, probably due to the proliferation of grid based and wavefront construction type schemes. Velis & Ulrych (1996) propose a fully non-linear shooting method of ray tracing that uses simulated annealing to locate the global minimum path in two dimensional velocity models. Simulated annealing (e.g. Kirkpatrick et al., 1983; Otten & van Ginneken, 1989) is based on an analogy with physical annealing in thermodynamics, where heating and controlled cooling of a material is used to increase the size of crystals and reduce their defects. Heating allows atoms to wander throughout the structure and slow cooling gives them time to find a configuration with lower internal energy than their initial state. Similarly, variation to the model parameters, in this case the initial ray trajectories, can be guided to a global minimum, given infinitely slow changes in temperature. Simulated annealing is a single perturbative approach for finding a single minimum per run. Its convergence to a global minimum can only be proven for infinitely slow changes in the temperature parameter, but in practice a finite step has to be used. Velis & Ulrych (2001) extend the method to three dimensions and implement a versatile model parameterization scheme. However their approach does not appear to be practical for finding all multipaths, and tends to be more computationally expensive than iterative non-linear solvers.

An alternative to ray shooting is to begin with an arbitrary initial path, and then iteratively adjust its geometry until it becomes a true ray path (i.e. it satisfies Fermat's principle of stationary time). A common approach to implementing this so called bending method is to derive a boundary value formulation of the kinematic ray tracing equations, which can then be solved iteratively (e.g. Julian & Gubbins, 1977). However, as in the shooting method the resulting ray path is not necessarily the first arrival ray path, as the technique cannot distinguish between a local an global extrema. This could be overcome if a fully non-linear search is used, but such an approach is likely to encounter similar limitations to the simulated annealing shooting scheme of Velis & Ulrych (1996, 2001). Pereyra et al. (1980) extend the ray bending technique so that interfaces can be included, and use shooting to obtain an initial ray path. They use a separate system of equations for each layer and couple them by applying the known discontinuity condition at each interface that is traversed by the ray path. In ray bending, a common method for obtaining a good initial guess for the ray path is to use the two point path from a laterally averaged version of the model (e.g. Thurber & Ellsworth, 1980; Sambridge & Kennett, 1990).

Ray tracing has been widely used in seismic tomography (see Iyer & Hirahara (1993) and Rawlinson & Sambridge (2003) for a comprehensive range of examples); specific applications include imaging the structure between boreholes (e.g. Bregman et al., 1989), the source area of earthquakes (e.g. Zhao et al., 1996a), subducting slabs (e.g. Conder & Wiens, 2006) and mantle upwelling (e.g. Toomey et al., 1998).

1.3 Grid based methods

1.3.1 First arrival schemes

A common alternative to tracing rays is to compute the first arrival travel time to all nodes of a grid which spans the medium (e.g. Vidale, 1988; van Trier & Symes, 1991; Kim & Cook, 1999; Sethian & Popovici, 1999). The resulting travel time field implicitly contains the wavefront location as a function of time (i.e. isochrons of the travel time field), and all possible first arrival ray paths are given by the gradient of the travel time field. Grid based methods have evolved to the point where many can guarantee to locate the first arrival travel time and ray path to all points of the medium (e.g. Rawlinson & Sambridge, 2004a; Buske & Kästner, 2004), even for highly heterogeneous media, where ray tracing is likely to fail. In figure 1.4, wavefronts have been computed using a finite difference eikonal solver for the same velocity model through which rays have been traced in figure 1.3. The wavefronts extracted from the travel time field appear to be stable despite the strong heterogeneity that causes significant focusing and defocusing of the rays.

The last 15 years has seen the development of numerous grid based algorithms for efficient computation of arrival times using various finite difference solutions of the eikonal equation. One of the first attempts to compute the first arrival travel time field using a finite difference technique was made in two dimensions by Vidale (1988) who later extended it to three dimensions (Vidale, 1990). The scheme involves progressively integrating travel times along an expanding square in two dimensions or an expanding cube in three dimensions. However, the use of an expanding square to the define the shape of the computational front cannot always respect the direction of flow of travel time information. Hole & Zelt (1995) implement an iterative post sweeping scheme to help account for the non causal nature of the expanding square. Such an idea was first introduced by Schneider et al. (1992). During post sweeping, the travel time field is recomputed several times in different directions, to account for changes in the flow of travel time information caused by velocity heterogeneity. For a two dimensional structure one iteration of the post sweeping procedure involves computing a travel time field starting at the four boundaries of the computational domain. At each grid node the minimum between the newly computed solution and the last updated solution is chosen. Kim & Cook (1999) apply post sweeping in their scheme until the travel time field converges (i.e. the values of the travel time do not change significantly in future applications of the post sweeping steps). They conclude that two applications of their post sweeping procedure leads to sufficiently accurate travel times in their examples.

Harten et al. (1987) and later Shu & Osher (1988, 1989) developed a uniformly higher order essentially non oscillatory scheme (ENO) for the numerical approximation of the viscous solution of a hyperbolic conservation law. The eikonal equation has the form of a hyperbolic conservation law and the first arrivals are

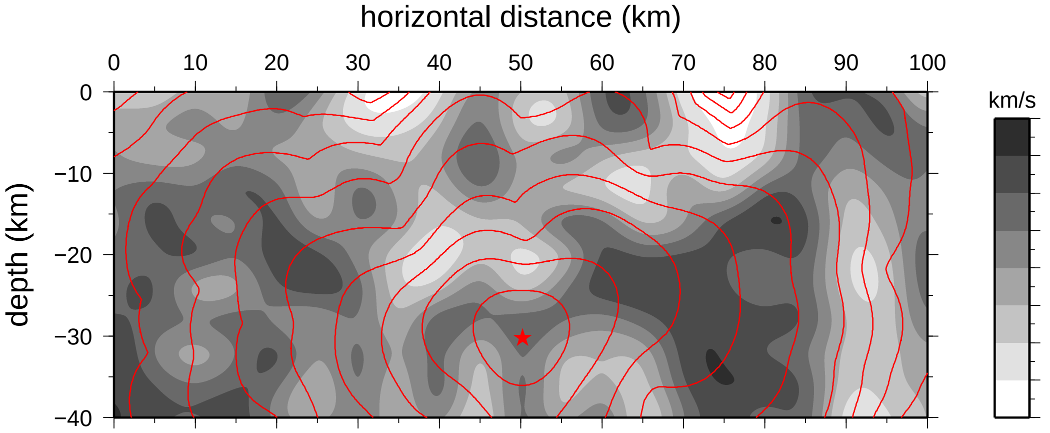


Figure 1.4: First arrival wavefronts calculated using an approach in which the travel time is computed along the boundaries of an expanding box with a fifth order WENO scheme and post sweeping. The algorithm used here is similar to the one described by Kim & Cook (1999). The velocity model is the same as in figure 1.3, and wavefronts are contoured at 1 s intervals.

therefore given by the viscous solution. The weighted essentially non oscillatory scheme (WENO) is a modification of the ENO scheme and has some advantages as far as computation time and stability are concerned (Liu et al., 1994; Jiang & Shu, 1996; Jiang & Peng, 2000).

Several grid based schemes for the computation of first arrival travel times are based on this class of numerical method. Qian & Symes (2002) solve the paraxial eikonal equation for a source at the surface and a wave propagating primarily in the depth direction using a third order WENO scheme and adaptive gridding. Solving the paraxial eikonal equation has the disadvantage that overturning waves cannot be properly represented. An overturning wave moves in a direction which differs from the propagation direction at the source point by more than 90 °. Kim & Cook (1999) solve the paraxial eikonal equation using a second order ENO scheme in space with a third order Runge Kutta scheme in time on the boundaries of an expanding box in three dimensions. They attempt to correct the travel times for overturning waves by applying post sweeping to the travel time field. Figure 1.4 shows wavefronts computed using a fifth order WENO scheme on an expanding box with post sweeping. The method can only predict first arrivals and therefore the wavefront develops kinks where swallowtails would otherwise form. Buske & Kästner (2004) compute travel times using an ENO scheme in polar coordinates with the origin co-located with the source. The advantage of this approach is that the wavefront in the source neighbourhood is not under sampled by the grid, and therefore has the potential to improve accuracy.

Another popular method for determining first arrival travel times at all points of a gridded velocity model is the shortest path method (e.g. Nakanishi, 1985; Moser, 1991; Cheng & House, 1996). In essence, this is an application of graph theory to the problem of ray tracing (e.g. Bondy & Murty, 1976). Seismic ray paths are computed by calculating the shortest travel time path through a network, which represents the velocity medium, using Dijkstra-like algorithms (Dijkstra, 1959). According to Fermat's principle the path taken between two points by a ray (e.g. light or a seismic wave) corresponds to an extremum (minimum or maximum) of the travel time. This means that the shortest time path between two points is a true ray path. In seismic imaging, shortest path ray tracing has been used by Nakanishi & Yamaguchi (1986) in local earthquake tomography, and Toomey et al. (1994) in the inversion of three dimensional refraction data. So called hybrid methods based on the graph method and ray bending have been used to image crustal structure (e.g. Korenaga et al., 2000). In this context the shortest path method is used to compute an initial ray path for a bending method.

Another technique for computing the travel time of first arrivals is the fast marching method (e.g. Sethian & Popovici, 1999) or FMM, which is an Eulerian (i.e. grid based) wavefront evolution method that solves the eikonal equation using upwind finite differences. When compared with the previous finite difference techniques, FMM is able to compute correct travel times for overturning rays and can be implemented with unconditional stability (Sethian & Popovici, 1999). Kim & Cook (1999) claim that their grid based scheme is also unconditionally stable. In the case of FMM, the unconditional stability comes from the use of upwind entropy satisfying operators which are well behaved in the presence of discontinuities in the first arrival travel time field, together with a narrow band evolution technique that always satisfies causality. FMM can be extended to handle interfaces and hence the computation of travel times for refracted and reflected waves (Rawlinson & Sambridge, 2004b; de Kool et al., 2006).

In fact, most grid based methods can be modified to track reflected and refracted phases. Reflections can be found by tracking a first arrival travel time field from both source and receiver to the entire interface. Fermat's principle of stationary time can then be applied to locate reflection points along the interface (e.g. Podvin & Lecomte, 1991). This scheme has the advantage that multiple reflection paths can be found for a single interface, but a travel time field needs to be computed for each receiver in addition to each source. An alternative approach is to reinitialise the computational front from the point of minimum travel time on the interface. As a consequence, separate travel time fields do not need to be computed for the receivers (e.g. Li & Ulrych, 1993; Rawlinson & Sambridge, 2004a,b), but it is no longer possible to compute multiple reflection paths.

As mentioned above, important advantages of grid based algorithms, and in particular FMM, compared to traditional ray tracing, are their computational efficiency, algorithmic simplicity, robustness and solution completeness, as long as the arrival time is single valued. If required, wavefronts and rays can be obtained a posteriori by either contouring the travel time field or following the travel time gradient from receiver to source, respectively (e.g. Rawlinson & Sambridge, 2004a; de Kool et al., 2006)

In the field of applied mathematics these schemes, where an actual travel time field is computed on a mesh of points, are commonly known as Eulerian methods (e.g. Sethian, 1999), as the propagation of the wavefront is described by computing its arrival time at the nodes of a fixed underlying grid. These grid based schemes have been used in a variety of seismological applications, and are particularly useful if large travel time datasets have to be computed, such as in seismic tomography (e.g. Rawlinson et al., 2006a,b) or the migration of coincident reflection sections (e.g. Gray & May, 1994; Bevc, 1997; Popovici & Sethian, 2002). However, if later arrivals are required, the methods discussed above are no longer appropriate.

1.3.2 Multi arrival schemes

The question of whether or not first arrivals are sufficient for imaging complex structures was posed soon after the appearance of first arrival finite difference techniques. In the context of exploration geophysics, Geoltrain & Brac (1993) conjectured that most of the wavefield energy is contained in later arrivals and therefore first arrival travel times are not sufficient to give a good migration image. There have been attempts to compute multi valued travel time fields using only a first arrival solver. However, these schemes often include a rather ad hoc procedure for dividing the computational domain into single valued subregions, followed by application of a first arrival solver in each subregion. The solutions for the different subregions are then superimposed to construct the multi valued travel time field (e.g. Fatemi et al., 1995; Benamou, 1999).

A more complete approach requires new grid based formalisms that have no intrinsic limitation on the nature of the advancing wavefront. One possibility is to take advantage of the work of Osher & Sethian (1988), who pioneered the field of interface tracking. A wavefront at a certain moment in time can be interpreted as an interface between points of a grid which have been crossed by the wavefront and points which have not yet been crossed (i.e. an iso-contour of the travel time field). Their idea was to replace ray tracing (i.e. solving ordinary differential equations for Lagrangian trajectories and other associated Lagrangian variables), with the computation of Eulerian variables, (i.e. the solution of partial differential equations on a grid). To do this they developed the level set method, which keeps track of an interface by expressing it as the zero level set (zero contour line) of a function representing the signed distance to the interface. A signed distance function is an Eulerian variable used to describe the position of an interface. Its value at a certain node of an underlying grid is given by the distance to the closest point on the interface. The sign is used to determine on which side of the interface the node is located. For a closed interface the signed distance function is typically defined to be negative for points inside and positive for points outside of the interface. A signed distance function is an implicit description of an interface; its position is given by the zero iso-contour line or zero level set of the signed distance function. This is similar to a travel time field where the wavefronts are given by the isochrons. The concept of a signed distance function is discussed in much more detail in section 2.2.2.

The interface can be implicitly tracked by numerically solving a partial differential equation that describes the evolution of the signed distance function on a grid. Interfaces cannot intersect each other, but they can merge like rain drops on a smooth surface. This means that it is not possible to describe a wavefront swallowtail using the level set method directly; only the first arrival part of the wavefront can be described. The level set method has become the state of the art algorithm for the description of the evolution of interfaces between fluids and gases in two dimensions (e.g. Mulder et al., 1992; Sussman et al., 1994; Chang et al., 1996). If an interface crosses each cell of a grid only once (e.g. like a flame front or the first arriving part of a wavefront) the level set method can be formulated much more efficiently as a boundary value problem, in which time becomes the dependent variable, leading to the fast marching method (Sethian, 1999), discussed in the previous section.

The state of a particle on a wavefront is characterised by its position vector $\mathbf{x}$ and slowness vector $\nabla\mathbf{T}$, the gradient of the travel time field (i.e. the wavefront normal). The space spanned by components of the position vector $\mathbf{x}$ is known as real space or normal space. One can also define a phase space, which is spanned by components of the position vector and slowness vector. Phase space is part of the Hamiltonian formulation of ray theory (e.g. Chapman, 1985; Lambaré et al., 1996). Rays in normal space are then replaced by bicharacteristics in phase space (e.g. Chapman, 1985). A three dimensional reduced phase space can be created from two dimensional normal space by simply using the direction θ of the local wavefront normal as the third coordinate. The wavefront is unfolded into a smooth curve, which is sometimes referred to as a bicharacteristic strip (Osher et al., 2002). One reason for referring to this phase space representation of the wavefront as a bicharacteristic strip is due to the fact that it is defined by the bicharacteristics of the eikonal equation, which are the phase space equivalent of the characteristics of the eikonal equation in real space, which correspond to rays. The advantage of phase space or reduced phase space is that when the wavefront self intersects and contains sharp corners, its corresponding bicharacteristics curve will be locally smooth and single valued.

In order to use the level set method to track multi arrival wavefronts, Osher et al. (2002) describe the bicharacteristic strip in reduced phase space as the intersection of the zero level sets of two three dimensional functions. Engquist & Runborg (2003) devise the so called segment projection method, in which the bicharacteristic strip is represented as a set of segments. This scheme can be viewed as a compromise between explicit wavefront tracking (see next section) and the level set method. The segments are evolved independently on individual grids and the connectivity between segments is handled by interpolation. A more recent study by Qian & Leung (2004) uses the level set method for the computation of multi valued travel times that satisfy the paraxial wave equation.

Fomel & Sethian (2002) use the Liouville formulation of the ray tracing equations, a system of time independent partial differential equations (referred to as escape equations), which can be solved numerically on a grid in reduced phase space. The solutions correspond to arrival times at the boundary from every point in the phase space domain. Multi arrival information such as wavefront geometry and two point travel times is extracted with post-processing.

Computing the multi valued geometrical solution of the eikonal equation has always been a driving force in the level set community (e.g. Osher et al., 2002; Fomel & Sethian, 2002; Cockburn et al., 2005), and these attempts by applied mathematicians stand in stark contrast to decades of ray tracing in seismology. Level set proponents often claim that their method is superior to a Lagrangian approach for the computation of multi arrival travel times due to the implicit representation of the wavefront (e.g. Osher et al., 2002). In these schemes, however, the spatial resolution of the wavefront is always controlled by the underlying grid.

To date no significant applications of these techniques have emerged in geophysics (Benamou, 2003) and no comparisons between Eulerian level set methods and Lagrangian wavefront trackers have been published in seismology. Wavefront tracking in phase space has also not been extensively investigated in seismology, particularly in applications outside the exploration field. In addition, many of the algorithms above have only been tested in relatively simple two or three dimensional media i.e. with smoothly varying velocities and restricted peak to peak amplitudes. Chapter 2 will therefore investigate the practicality of the scheme proposed by Osher et al. (2002) for the computation of multi valued travel times.

1.4 Wavefront tracking

In this work wavefront tracking refers to schemes in which the wavefront is described explicitly (i.e. by a set of points and not as the isochron of a travel time field). Lagrangian approaches to the problem of seismic wavefront tracking were introduced in two dimensions by Lambaré et al. (1992) and Vinje et al. (1993) and in three dimensions by Vinje et al. (1999). In a Lagrangian method, the goal is to track the evolving wavefront explicitly using a set of points which describe the wavefront surface as opposed to implicit wavefront tracking where a fixed underlying grid of nodes is used (i.e. an Eulerian method). The basic principle is that a wavefront can be evolved by repeated applications of local ray tracing to a set of points lying on the wavefront. New points can be interpolated at each step to overcome the under sampling problems that may arise as the wavefront expands and distorts due to velocity heterogeneity. Redundant points could also be removed to improve efficiency, but to date, no published wavefront tracking scheme has implemented such a procedure.

Using initial value ray tracing, which is highly accurate, means that the main source of error for the location of the wavefront is the interpolation scheme. In practice, one is usually interested in the arrival time and ray path for a given receiver. This means that a scheme for interpolating an arrival time at a receiver from a set of wavefronts needs to be formulated.

In order to maintain a predetermined node density on each wavefront, new points can be added based on their separation distance (e.g. Vinje et al., 1993;

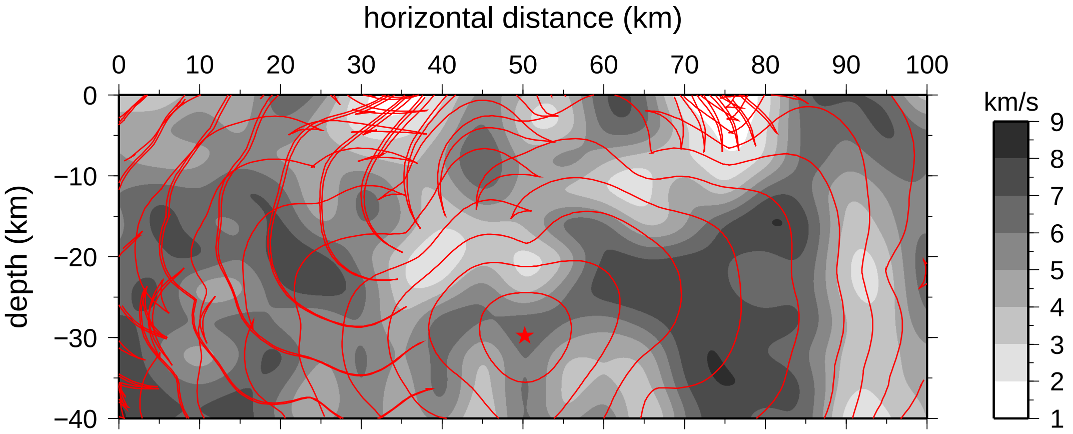


Figure 1.5: Multi arrival wavefront tracking using the Lagrangian approach presented in chapter 3. The wavefronts are contoured at 1 s intervals.

Ettrich & Gajewski, 1996; Leidenfrost et al., 1999). The problem with using a refinement criterion based on metric distance is that it does not account for variations in wavefront curvature, so regions of high detail are likely to be under sampled. Sun (1992) recognised that the angular distance and therefore curvature should also be taken into account when adding points to the wavefront. Vinje et al. (1996b) use the distance between two points on the wavefront and the angle between the corresponding wavefront normal as a refinement criterion. In three dimensions, a set of triangles (Vinje et al., 1996b) or squares (Gibson Jr et al., 2005) can be used to describe the connectivity between nodes on the wavefront.

Compared with the previously introduced grid based methods like the fast marching method and shortest path ray tracing, Lagrangian wavefront tracking has the advantage that it can be used to calculate later arrivals in addition to first arrivals. This is illustrated in figure 1.5, where the evolving wavefront develops several swallowtails. By comparing the wavefronts extracted from the first arrival travel time field (figure 1.4) with the multi arrival wavefronts in figure 1.5 it is clear that the first arrival segments of the wavefronts only partially describe the geometric dissipation of seismic energy. The later arriving swallowtails gradually expand in size as time progresses and the associated rays turn away from their initial direction by up to 180 ° in some cases (cf. green ray in figure 1.3).

Difficulties can still arise due to the presence of sharp corners, which usually occur when a wavefront triplicates. Most interpolation schemes are described in terms of smooth differentiable functions, and therefore may not produce satisfactory results in such circumstances. Lambaré et al. (1996), in two dimensions, and

Lucio et al. (1996), in three dimensions, use the Hamiltonian formulation of ray theory in full phase space for their wavefront tracking scheme. Using a full phase space or reduced phase space to represent the wavefront means that the bicharacteristic strip is locally smooth and will not self intersect even if the wavefront triplicates.

In the field of exploration geophysics multi valued travel times and amplitude maps obtained by wavefront tracking have been used for the migration of reflectors in complex two dimensional (e.g. Ettrich & Gajewski, 1996; Xu & Lambaré, 2004) and three dimensional media (Xu et al., 2004). However, it is worth noting that wavefront tracking has to date not been used in any solid earth applications, such as passive source tomography, the prediction of global phases, and so forth.

Using the expression *Eulerian scheme* to refer to grid based methods and *Lagrangian scheme* to refer to methods based on explicit wavefront tracking is not common in seismology. The terminology is, however, widely used in the field of applied mathematics in reference to interface evolution techniques (e.g. Sethian, 1999; Osher et al., 2002; Osher & Fedkiw, 2003; Benamou, 2003). For convenience, and to acknowledge the important contributions made by applied mathematicians to this field of research, this terminology will be used in the following dissertation (see appendix A for a glossary).

1.5 Outline

In chapter 2 a level set method is developed for the computation of multi valued travel times. The scheme has previously been suggested by Osher et al. (2002), but so far has not been used in practical seismological problems. In any Eulerian approach, the spatial resolution of a wavefront (i.e. an isochron of the travel time field) is limited by the resolution of the underlying grid. It will be shown that for multi valued wavefront tracking in reduced phase space, the grid resolution imposes severe restrictions on the level of detail that can be retained during the propagation process.

The alternative approach of explicit wavefront tracking in reduced phase space using a Lagrangian scheme is presented in chapter 3. Wavefront tracking in real space has previously been suggested by Vinje et al. (1993) and is used in exploration geophysics. The new scheme will turn out to be much more suited to the computation of multi valued travel times than the Eulerian technique.

The Lagrangian scheme is extended in chapter 4 so that wavefronts can be tracked in the presence of interfaces, which also give rise to later arrivals of reflected and refracted waves. Since identifying later arrivals in observations is a major issue, the Gaussian beam method is used for the computation of ray based seismograms, which could potentially help with the identification of later arrivals.

One of the goals in developing a scheme for the computation of later arrivals is to investigate how it might be used in seismic imaging. Chapter 5 explores the potential benefits and pitfalls of incorporating multi arrival travel times in seismic tomography. It will be shown that later arrivals can indeed improve the quality of results obtained by seismic imaging, but they also tend to make the inverse problem more non-linear.

The focus in this thesis up to and including chapter 5 is on later arrivals in two dimensional models. In chapter 6, wavefront tracking in three dimensional models using the full six dimensional phase space is discussed. Concepts for evolving surfaces first proposed in the field of computer graphics will be used here to describe propagating wavefronts. The results demonstrate that the proposed new technique is sufficiently stable for application to highly complex models.

Finally a summary and conclusions of the work presented in this thesis is given in chapter 7, where avenues for future work are also discussed. The dynamic of an evolving wavefront is often best observed in an animation. Appendix E therefore contains a CD with movies of the propagating wavefronts described in several examples presented in this work.

Chapter 2

Eulerian scheme

Propagating interfaces occur not only in seismic wavefront tracking but also in a wide variety of other settings, and include ocean waves, crystal growth, flame fronts and material boundaries (e.g. in fluid mechanics). The perspective on seismic wavefront tracking presented here emanates from a large and rapidly growing body of work which relies on a grid-based finite difference approach for computing interface evolution (e.g. Sethian, 1999; Osher & Fedkiw, 2003). These methods have only recently been suggested as an alternative to Lagrangian wavefront tracking (e.g. Sethian & Popovici, 1999; Fomel & Sethian, 2002; Osher et al., 2002; Qian & Leung, 2004). At their core lie two computational techniques: fast marching methods and level set methods. Both are characterised by a fundamental shift in how one views moving boundaries. They rethink the Lagrangian geometric perspective and replace it with the finite difference solution of a partial differential equation on a grid. A wavefront is then defined as a contour line or surface of a discrete travel time field (Eulerian), and no longer by a set of points (Lagrangian).

In this chapter the initial value partial differential equation which describes interface motion is first formulated. This will lead to the concept of a signed distance function and eventually to the level set equation and its viscous solution. A reduced phase space is defined in order to track a self-intersecting wavefront. Finally, the method is used to calculate wavefronts for a constant velocity model, a wave guide model and a subduction zone setting.

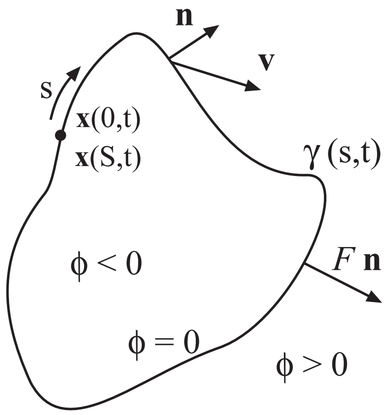


Figure 2.1: Interface propagation under the influence of a velocity field $\mathbf{v}$. The resultant motion in the outward normal direction is defined by the speed function $F = \mathbf{n} \cdot \mathbf{v}$.

2.1 Formulation of interface propagation

Consider a boundary - a curve in two dimensions or a surface in three dimension - separating one region from another. The motion of this boundary is defined by a known speed function F in a direction normal to itself, where the normal direction is oriented outward with respect to a pre-defined inside and outside. The boundary does not necessarily have to be a closed boundary as long as the orientation of the speed F with respect to the two possible normal directions has been defined. The goal is to track the motion of the interface as it evolves. Only the motion of the interface in its normal direction is considered; motions in the tangential directions are ignored. The speed function can depend on many factors including the position of the boundary, its curvature and normal direction.

Given a simple, smooth, closed initial curve γ in $\mathbf{R}^2$ and the family of curves $\gamma(t)$ generated by moving γ along its normal vector field with speed F, a natural approach is to parameterize γ at time t using the position vector $\mathbf{x}(s,t)$, where s is the path length along the interface (see figure 2.1). The total length of the front is given by S with $0 \leq s \leq S$ which means that $\mathbf{x}(0,t) = \mathbf{x}(S,t)$. This is a Lagrangian formulation because $\mathbf{x}(s,t)$ describes the moving front explicitly.

Under the assumption that $F > 0$ the front always moves outward. An alternative way to characterise the position of this expanding front is to calculate the arrival time $T(\mathbf{x})$ of the front as it crosses each point of an underlying grid. The equation for this arrival time function $T(\mathbf{x})$ is then given by

$$|\nabla T| F = 1, \qquad T = 0 \text{ on } \Gamma, \tag{2.1}$$

where Γ is the initial position of the interface and ∇T is the gradient of the arrival time field orthogonal to the front. The motion of the front is now described by the solution to a boundary value problem, which is an Eulerian formulation because the front is given by the contour line of the arrival time field which has been defined on a grid. This boundary value formulation eventually gives rise to the fast marching method. If the speed F depends only on the position (i.e. $F = F(\mathbf{x})$) then (2.1) reduces to what is known as the eikonal equation in seismology.

The eikonal equation is the so-called high frequency approximation of the full elastic wave equation (e.g. Aki & Richards, 2002). It is derived from the elastic wave equation under the assumption that the wavelength of the propagating wave is substantially shorter than the seismic heterogeneities it encounters (e.g. Červený, 2001; Chapman, 2004). It is normally given as (Chapman, 2004)

$$|\nabla T| = s \tag{2.2}$$

where $s = 1/F$ is slowness and T is a time function (the eikonal) which describes surfaces of constant phase (wavefronts) when T is constant.

However, if the speed F is permitted to vary in sign the front can move forward or backward and hence may pass over a point $\mathbf{x}$ of the underlying grid several times. In this case the crossing time or arrival time $T(\mathbf{x})$ is no longer a single valued function. One way of accounting for this added complexity is to define the initial position of the front as the zero level set (zero contour line) of a higher dimensional function ϕ (i.e. a signed distance function). The evolution of this function ϕ can then be linked to the propagation of the interface through a time dependent initial value problem where the position for a given time corresponds to the zero level set of ϕ. This initial value formulation eventually leads to the level set method.

2.2 Level set method

As discussed in the previous section, an interface can be described as the zero level set of a higher dimensional function. Instead of evolving the interface directly, the higher dimensional function is updated in time and when the position of the interface is required, the zero contour line or zero level set of this function is extracted.

2.2.1 Level set equation

The aim now is to derive an equation of motion for the level set function ϕ by matching the zero level set of ϕ with the evolving front. The level set value of a particle $\mathbf{x}(t)$ moving along with the front must always be zero, and therefore

$$\phi(\mathbf{x}(t), t) = 0. \tag{2.3}$$

The time derivative of (2.3) is found with the chain rule

$$\phi_t + \nabla\phi(\mathbf{x}(t), t) \cdot \mathbf{x}_t(t) = 0, \tag{2.4}$$

where ϕ_t and $\mathbf{x}_t(t)$ denote derivatives with respect to time. Since the scalar speed function F is defined as the speed in the outward normal direction, one can write

$$F = \mathbf{x}_t(t) \cdot \mathbf{n}, \tag{2.5}$$

where $\mathbf{n}$ is the unit normal vector to the interface defined by

$$\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}. \tag{2.6}$$

Substituting (2.6) into (2.5) yields $F|\nabla\phi| = \nabla\phi \cdot \mathbf{x}_t(t)$ and from (2.4) the evolution equation for the signed distance function ϕ is:

$$\phi_t + F|\nabla\phi| = 0, \quad \text{given} \quad \phi(\mathbf{x}, t = 0). \tag{2.7}$$

This is the level set equation formulated by Osher & Sethian (1988), and corresponds to equations describing advective transport in an incompressible fluid. From a conservation law point of view (2.7) states that for a given point the change of a property with time (for example the concentration of a substance in a fluid) is equal to the flux of this property in the direction of the gradient. If the speed function F depends on the curvature of the interface, i.e. on the second derivative of the signed distance function, the level set equation becomes what is known as a hyperbolic conservation law.

The level set equation has been introduced for the motion of the front in the outward normal direction at the speed $F(\mathbf{x})$. However, it is possible to modify this equation so that it describes the evolution of an interface under the influence of a velocity $\mathbf{v}(\mathbf{x})$, which is a vector field. The level set equation (2.7) then becomes

$$\phi_t + \mathbf{v} \cdot \nabla \phi = 0, \quad \text{given} \quad \phi(\mathbf{x}, t = 0). \tag{2.8}$$

The velocity $\mathbf{v}$ in (2.8) or the speed F in (2.7) may depend on many factors. For example, F can depend on local geometrical information associated with front curvature, or the direction of the local wavefront normal. In fluid dynamics it is common to encounter an interface evolving under curvature dependent motion (e.g. Osher & Sethian, 1988; Evans & Spruck, 1991). Examples of local properties influencing the evolution of an interface are bubble dynamics and two phase flow where surface tension plays a role (e.g. Sussman et al., 1994; Chang et al., 1996). F can also be influenced by the global properties of a front (i.e. its shape and position), and might depend on integrals along the front and/or associated differential equations. A particular example of this occurs when an interface is a source of heat that affects diffusion on either side of the interface which, in turn, affects the motion of the interface (e.g. Ruuth, 1998). In seismic wavefront tracking, the speed of the front is a function of position $\mathbf{x}$, and does not depend on the shape of the front. It is also possible to define a speed which depends on more than one of these properties, for example, a gas bubble rising in a moving fluid.

2.2.2 Signed distance function

A natural choice for the implicit representation of a curve in two dimensions or a surface in three dimensions is a signed distance function. For each node of the underlying grid, the distance to the closest point on the curve or surface is calculated. The distance is negative for points inside of the front and positive for points outside of the front (see figure 2.2 for an example in two dimensions).

Such a signed distance function ϕ is well behaved when the absolute value of the gradient is equal to one for every point in the computational domain. In this situation (2.7) reduces to $\phi_t = -F$ and the values of ϕ either increase or decrease, depending on the sign of F. When $F > 0$ the interface moves in the outward normal direction, and when $F < 0$ the interface moves in the inward normal direction. When $F = 0$ the equation reduces to $\phi_t = 0$ and hence the interface does not move.

For points that are equidistant from at least two points on the front, the gradient of ϕ is discontinuous and hence the signed distance function is no longer well behaved. This means that any numerical method used for evolving the signed

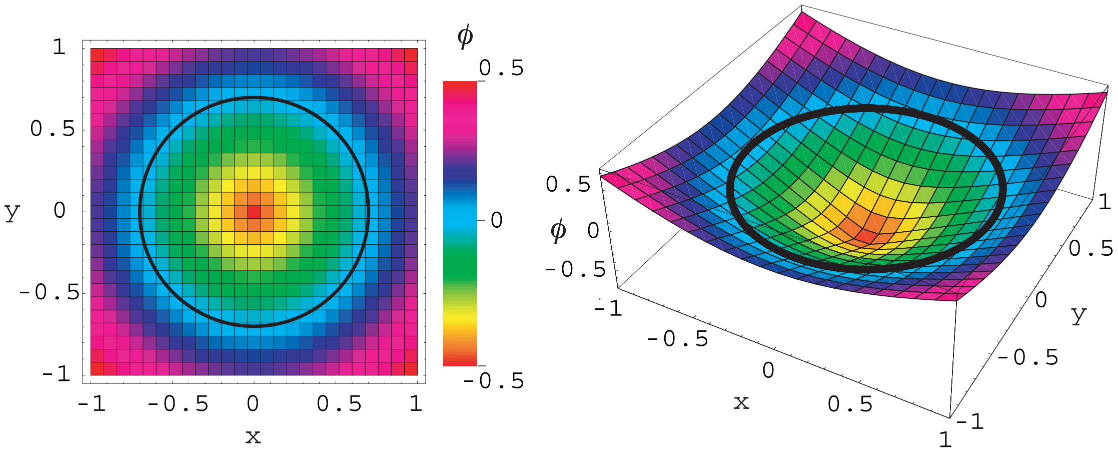


Figure 2.2: Signed distance function ϕ used to implicitly define a circle with radius 0.75 and centre $(0, 0)$.

distance function has to exhibit a reasonable behaviour at the occasional kinks where $|\nabla\phi|$ is not defined.

The concept of a signed distance function requires that there is an inside and outside with respect to the interface. If the interfaces have multiple junctions, for example a network of soap bubbles or grain boundaries in a rock, the idea is to assign to each region (i.e. individual soap bubbles or grains) a separate signed distance function to describe its boundary (e.g. Merriman et al., 1994; Zhao et al., 1996b). The signed distance functions are then evolved independently, which means that gaps between the regions (i.e. soap bubbles or grains) are likely to develop. In a so called interaction step, the signed distance functions assigned to the different regions can be updated so that the junction points (i.e. where boundaries meet) show the desired behaviour (Merriman et al., 1994).

2.2.3 Basic algorithm for interface evolution

A key step in the development of the level set method was the realisation that schemes from computational fluid mechanics, specifically designed for approximating the solution to hyperbolic conservation laws, can be used to solve the level set equation. This idea was developed by Sethian (1987) and is at the core of the level set method. To understand the strategy, schemes used for approximating hyperbolic conservation laws have to be introduced. These schemes will be illustrated in the following by using a one dimensional model.

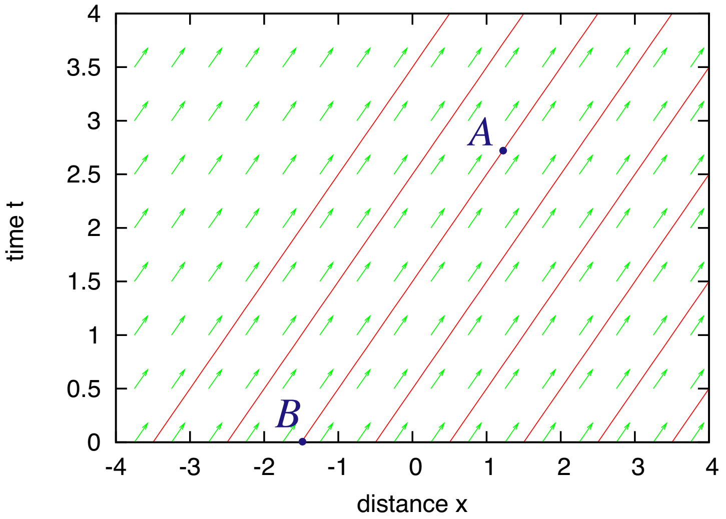


Figure 2.3: Characteristic curves (red lines) for $F = 1$ in the x-t plane. u is constant along lines with slope 1.

2.2.3.1 Solution in one dimensional space

Consider the level set equation for a one dimensional problem, where the signed distance function is given by u:

$$u_t(x,t) + Fu_x(x,t) = 0 \quad \text{with} \quad u(x,0) = f(x), \tag{2.9}$$

where u_t is the time derivative of u, u_x the spatial derivative and F the speed. Although it still can be viewed as a conservation law, (2.9) is also the one dimensional wave equation, and for constant F a solution is given by $u(x,t) = f(x-Ft)$. This means that the solution u at any point x at time t is given by the value of the initial data at the point $x - Ft$ on the x axis. In addition, the solution u is constant along lines of slope F in the x-t plane (figure 2.3). Considering two points A and B in the x-t plane the solution at point A can be found by tracing back along a line with slope F to the point B on the x-axis. Hence, in formal terms, the domain of dependence of the point A is the point B. Conversely, the set of points on the line with slope F emanating from point B is referred to as the domain of influence of point B. The lines of constant u in the x-t plane are known as characteristics, and more specifically, those in figure 2.3 are the characteristics of the one dimensional wave equation (2.9) with $F = 1$.

The standard approach (Sethian, 1999) for numerically solving (2.9) is to discretise x-t space into a collection of grid points with a spatial spacing of Δx and

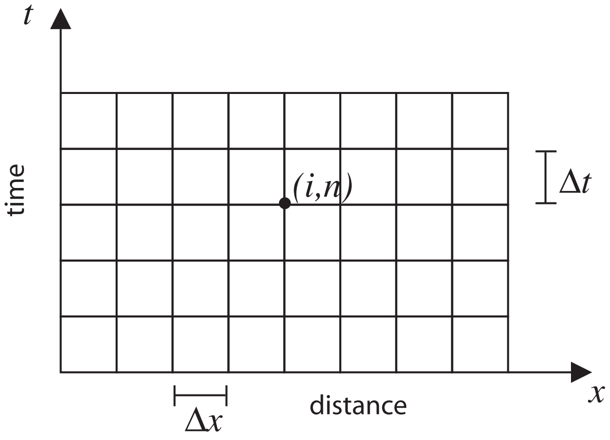


Figure 2.4: Computational grid used to solve (2.9).

a time step of Δt (figure 2.4). Every grid point can then be represented by a coordinate pair (i, n) corresponding to the point $(i\Delta x, n\Delta t)$. The time derivative u_t in (2.9) can be approximated by a forward finite difference operator,

$$u_t = \frac{u_i^{n+1} - u_i^n}{\Delta t}. \tag{2.10}$$

Substituting the forward finite difference operator (2.10) for u_t in (2.9) produces the following expression for u_i^{n+1},

$$u_t^{n+1} = u_t^n - \Delta t u_x F, \tag{2.11}$$

which can be used to approximate u ahead in time once the spatial derivative u_x is calculated. The spatial derivative can be discretised using a forward $\Delta^+ u_x$, backward $\Delta^- u_x$ or centred $\Delta^0 u_x$ finite difference operator. These operators are given as

$$\Delta^+ u_x = \frac{u_{i+1}^n - u_i^n}{\Delta x}, \tag{2.12}$$

$$\Delta^- u_x = \frac{u_i^n - u_{i-1}^n}{\Delta x}, \tag{2.13}$$

$$\Delta^0 u_x = \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x}. \tag{2.14}$$

While the centred operator is a more accurate approximation, the scheme which should be used depends on the previous discussion about domains of dependence and characteristics. For $F = 1$, information travels from left to right in figure 2.3.

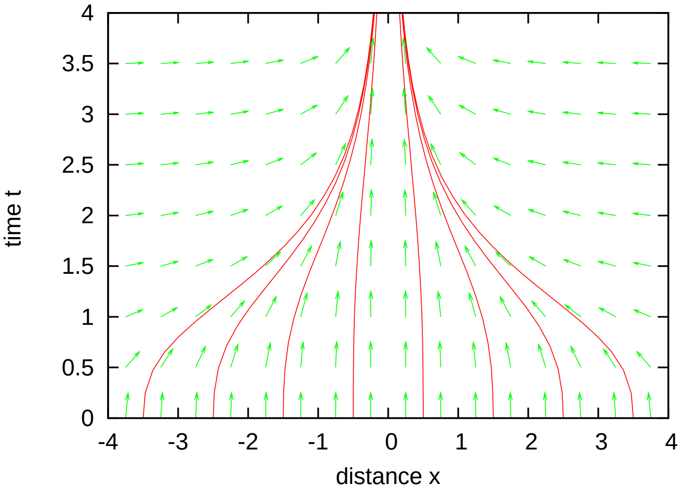


Figure 2.5: Characteristics of the one dimensional wave equation where F is a function of x and t given by $F(x,t) = -\frac{2}{5}x(t+\frac{1}{4})$.

The backward scheme (2.13) in this situation is referred to as an upwind scheme because it uses values upwind of the direction of information propagation, which is clearly preferable, as it sends information in the direction that correctly matches the differential equation. Generally speaking, the numerical domain of dependence should contain the mathematical domain of dependence. One commonly refers to the set of nodes used for the computation of a spatial derivative at a certain node as a stencil.

If F is function of x and/or t, the characteristics are no longer given by a set of parallel straight lines (see figure 2.5). Instead lines may converge and form what is known as a shock in fluid dynamics, or diverge and form what is known as a rarefaction. Hence, which finite difference operator is chosen for the approximation of u_x at the nodes along x for a given time t, will depend on the local direction of information flow, so that the numerical domain of dependence always includes the mathematical domain of dependence.

The numerical approximation of conservation laws and especially hyperbolic conservation laws relies on schemes which consider the flow of information when approximating spatial derivatives. As the goal here is to understand a basic technique for approximating a single hyperbolic conservation law, the focus will be on the so called Engquist-Osher or EO scheme (Engquist & Osher, 1980). While there are several alternatives, among them the Lax-Friedrich scheme and Riemann solvers, the so called EO scheme, which is based on Godunov's scheme (Godunov, 1959), is one of the simplest and most common schemes in the level set field.

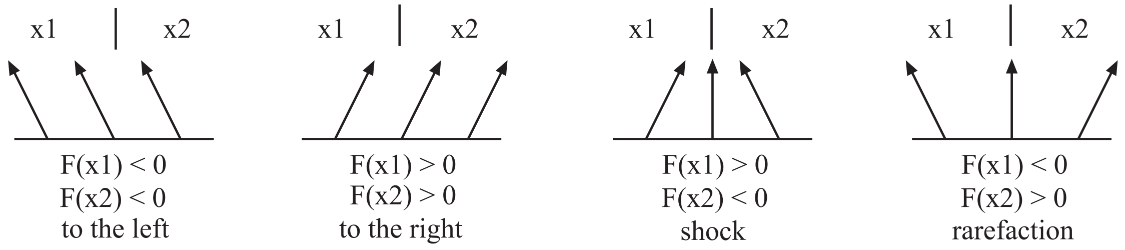


Figure 2.6: Possible configurations of characteristics around a point in the x-t plane.

Figure 2.6 shows all possible cases for the directions in which information is sent from a node of the computational grid at a given time.

1. $F(x1), F(x2) < 0$: The wave moves to the left and information is sent from right to left. Therefore the forward finite difference operator should be used.
2. $F(x1), F(x2) > 0$: The wave moves to the right and information is sent from left to right. Therefore the backward finite difference operator should be used.
3. $F(x1) > 0, F(x2) < 0$: Two waves collide and a shock develops, which moves with the sum of the two speeds, since the speed on the right is negative (the characteristics move to the left) and speed on the left is positive (characteristics move to the right).
4. $F(x1) < 0, F(x2) > 0$: The wave is split in two and a rarefaction develops, which does not move, since the speed on the right side is positive (characteristics go to the right), while the speed on the left is negative (characteristics move to the left).

One can then directly write down a finite difference scheme for u_x, which chooses the correct operator for all four cases:

$$u_x = \sqrt{\max(\Delta^- u_x, 0)^2 + \min(\Delta^+ u_x, 0)^2} \tag{2.15}$$

The resulting scheme for updating a discretised signed distance function in $\mathbf{R}^1$ is then given as

$$u_i^{n+1} = u_i^n - \Delta t \sqrt{\max(\Delta^- u_x, 0)^2 + \min(\Delta^+ u_x, 0)^2}, \tag{2.16}$$

with $\Delta^- u_x$ and $\Delta^+ u_x$ given by (2.12) and (2.13) respectively. This solution of a hyperbolic conservation law is generally known in the field of applied mathematics as a viscous solution or minimum entropy solution; by taking the flow of information into account the numerical scheme stays well behaved. In the following section the scheme will be extended to higher spatial dimensions.

2.2.3.2 Solution in a higher dimensional space

The level set equation (2.7) for three spatial dimensions can be written in the form of the general Hamilton-Jacobi equation

$$\phi_t + H(\phi_x, \phi_y, \phi_z, x, y, z) = 0, \tag{2.17}$$

where H is known as the Hamiltonian, which for the level set equation (2.7) is given by

$$H(\phi_x, \phi_y, \phi_z, x, y, z) = F\sqrt{\phi_x^2 + \phi_y^2 + \phi_z^2}. \tag{2.18}$$

A one dimensional version of (2.17) can be written

$$\phi_t + H(\phi_x) = 0. \tag{2.19}$$

Partial differentiation of (2.19) with respect to x reveals

$$\frac{\partial \phi_x}{\partial t} + [H(\phi_x)]_x = 0. \tag{2.20}$$

Substituting $u = \phi_x$ gives the hyperbolic conservation law

$$u_t + [H(u)]_x = 0. \tag{2.21}$$

This means that the viscous solution of the level set equation, as introduced in the previous section, can also be computed using schemes proposed for general Hamilton-Jacobi equations (Sethian, 1999). This link is useful, because there exists a wide variety of higher order numerical solvers for general Hamilton-Jacobi equations, which are known to converge to the viscous solution (e.g. Osher & Sethian, 1988; Jiang & Peng, 2000; Zhang & Shu, 2003; Bryson & Levy, 2003), and therefore can be used here.

Although such a relationship between the hyperbolic conservation law and the general Hamilton-Jacobi equation does not exist for more than one dimension, higher dimensional numerical schemes can nevertheless be constructed by using a dimension by dimension extension of the basic one dimensional approach. Schemes for higher dimensional Hamilton-Jacobi equations for symmetric Hamiltonians can be built by simply replicating each spatial variable. This allows a finite difference scheme for the level set equation to be built for any given m dimensional space. By applying (2.16), a scheme for three spatial dimensions is given by (Osher & Sethian, 1988)

$$\phi_{i,j,k}^{n+1} = \phi_{i,j,k}^{n} - \Delta t[\max(F_{i,j,k}, 0)\nabla^{+} + \min(F_{i,j,k}, 0)\nabla^{-}], \tag{2.22}$$

where

$$\begin{aligned}\nabla^{+} = [\,&\max(\Delta^{-}\phi_x, 0)^2 + \min(\Delta^{+}\phi_x, 0)^2 +\\ &\max(\Delta^{-}\phi_y, 0)^2 + \min(\Delta^{+}\phi_y, 0)^2 +\\ &\max(\Delta^{-}\phi_z, 0)^2 + \min(\Delta^{+}\phi_z, 0)^2]^{\frac{1}{2}}\end{aligned} \tag{2.23}$$

and

$$\begin{aligned}\nabla^{-} = [\,&\max(\Delta^{+}\phi_x, 0)^2 + \min(\Delta^{-}\phi_x, 0)^2 +\\ &\max(\Delta^{+}\phi_y, 0)^2 + \min(\Delta^{-}\phi_y, 0)^2 +\\ &\max(\Delta^{+}\phi_z, 0)^2 + \min(\Delta^{-}\phi_z, 0)^2]^{\frac{1}{2}}.\end{aligned} \tag{2.24}$$

$\Delta^{-}\phi_x$, $\Delta^{-}\phi_y$ and $\Delta^{-}\phi_z$ are the first order backward finite difference operators in the x, y and z direction, and $\Delta^{+}\phi_x$, $\Delta^{+}\phi_y$ and $\Delta^{+}\phi_z$ are the corresponding forward first order finite difference operators. If the interface moves under the influence of a velocity $\mathbf{v}$ (see (2.8)) instead of the speed F in the outward normal direction (see (2.7)), the scheme for three spatial dimensions is as follows:

$$\begin{aligned}\phi_{i,j,k}^{n+1} =& \phi_{i,j,k}^{n} - \Delta t\\ &[\max(v_{i,j,k}^{x}, 0)\Delta^{-}\phi_x + \min(v_{i,j,k}^{x}, 0)\Delta^{+}\phi_x\\ &\ \max(v_{i,j,k}^{y}, 0)\Delta^{-}\phi_y + \min(v_{i,j,k}^{y}, 0)\Delta^{+}\phi_y\\ &\ \max(v_{i,j,k}^{z}, 0)\Delta^{-}\phi_z + \min(v_{i,j,k}^{z}, 0)\Delta^{+}\phi_z],\end{aligned} \tag{2.25}$$

where $v_{i,j,k}^{x}$, $v_{i,j,k}^{y}$ and $v_{i,j,k}^{z}$ are the x, y and z components of the velocity vector $\mathbf{v}$ at the point (i, j, k).

The level set method can generally be used to evolve an $m-1$ dimensional manifold in an m dimensional space. A line is a one dimensional manifold and a surface a two dimensional manifold. This means that one can evolve a line in two

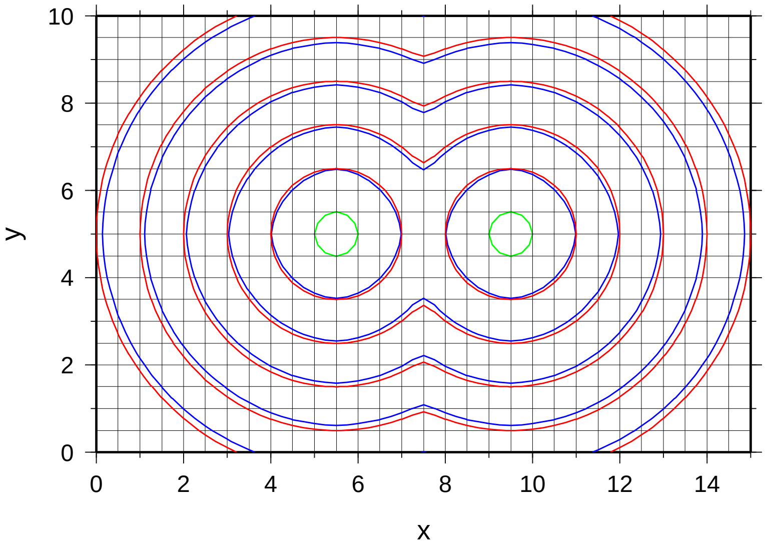


Figure 2.7: Two interfaces start off as a circles (green lines) and propagate with a constant speed of one in their outward normal direction. They merge and form one new interface. The interfaces are calculated using a first order accurate scheme (blue lines) and a fifth order WENO scheme (red lines) and plotted at 1 s intervals; interface segments parallel to the grid lines should therefore coincide with them.

dimensions or a surface in three dimensions. The first order scheme (2.22) will now be used to describe the evolution of an interface in two spatial dimensions. This means that derivatives with respect to z and the corresponding discretisation k are simply omitted in (2.22), (2.23) and (2.24).

In figure 2.7 two interfaces begin as circles of radius 0.5 (shown in green). The grid for the signed distance function consists of 50×75 nodes and the time step is 0.125 s. The speed F is defined in the outward normal direction and is equal to one. The interfaces are extracted after having up-sampled the signed distance functions using a cubic B-spline approximation. Due to the extraction of the zero level set contours at 1 s intervals, interface segments parallel to the grid lines in figure 2.7 should coincide with them. The interfaces computed using the first order method previously introduced are plotted in blue, while the red interfaces were computed using a higher order solver which will be introduced in the next section. Clearly the higher order approximation is preferable.

The interfaces in figure 2.7 do not intersect each other; instead they merge to form one new interface. Their behaviour mimics that of growing water drops on a surface. The attractiveness of the level set method for interface tracking lies in the implicit representation of the interface. There is no need to redefine the front during the evolution. Topological changes like breaking and merging

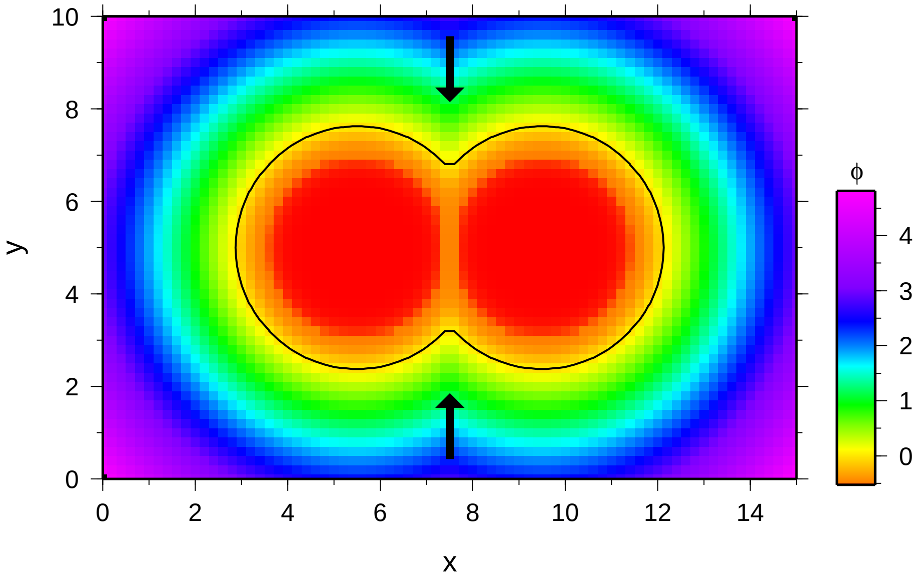


Figure 2.8: Signed distance function for one time step of the merging circle example. The line along which the signed distance function is discontinuous is indicated by the black arrows. The interface is given by the black line. The discontinuity has been preserved by using a fifth order WENO scheme for the approximation of the spatial derivatives.

of the evolving boundary are handled naturally. This explains why the level set method has been widely used in the field of fluid dynamics (e.g. Mulder et al., 1992; Sussman et al., 1994; Chang et al., 1996).

It is worthwhile remembering at this stage that a seismic wavefront becomes self-intersecting as it develops a swallowtail pattern in the presence of a low velocity anomaly (figure 1.1). Hence the level set method cannot be used directly to describe the wavefront. It will be shown later that it still can be used for wavefront tracking based on the concept of reduced phase space (see section 2.3), which allows the unfolding of a wavefront so that it no longer self-intersects (Osher et al., 2002). For the moment, however, the focus will be on further investigation of the level set method.

2.2.3.3 Higher order finite difference operator

Instead of the first order accurate spatial finite difference operators (2.12) and (2.13), one can use a higher order operator in order to gain additional accuracy. As outlined before, a signed distance function might contain discontinuities where a grid node is equidistant to two or more points on the interface. A potential drawback of simple higher order finite difference operators (e.g. (2.14)) is that they

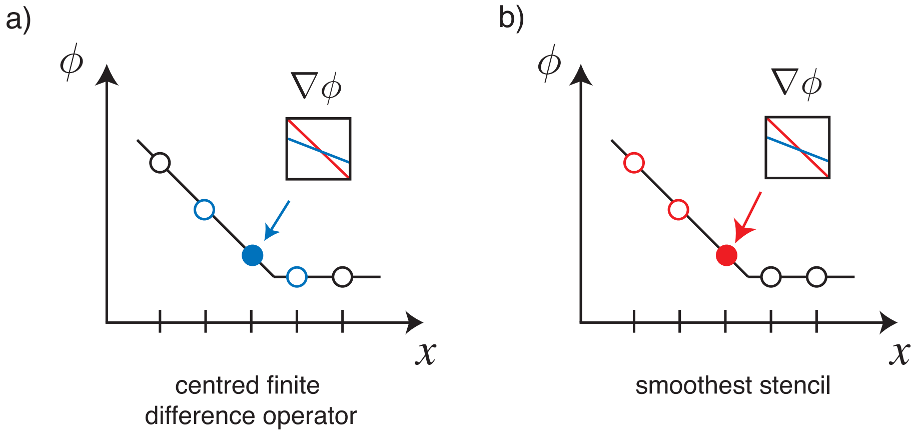


Figure 2.9: The gradient of a signed distance function is computed for a node (filled circle) near a discontinuity using two different finite difference operators. The two different gradients are plotted in the black boxes. (a) The stencil uses nodes on both sides. (b) The stencil is based on the smoothest set of nodes. Clearly only the scheme used in (b) can recover the correct gradient (red line).

tend to smooth out discontinuities in the signed distance function if their stencil includes a discontinuity like the one shown in figure 2.8. Here, a discontinuity in the signed distance function is visible along the line, indicated by the black arrows, along which the circles are merging. If it is important to represent this merging accurately, one has to avoid a smearing out of the discontinuity of the signed distance function.

The ideal approach is to use higher order accurate schemes designed for piecewise smooth functions containing isolated discontinuities. ENO (essentially non oscillatory) and WENO (weighted essentially non oscillatory schemes) are such schemes, and have been developed with hyperbolic conservation laws and related Hamilton-Jacobi equations in mind. ENO schemes were first proposed by Harten et al. (1987). Their idea was to start with one or two nodes and then add one node at a time to the stencil from the two neighbour candidates to the left and right. The node which provides the smoother stencil is then chosen to be added to the stencil.

Figure 2.9 illustrates in one dimension how a higher order finite difference operator smears out a discontinuity if it is not based on the smoothest set of nodes. In figure 2.9a a centred finite difference operator is used to compute $\nabla\phi$ for a node on the left side of a discontinuity (filled circle). The computed gradient is not correct as a node on the other side of the discontinuity is included in the stencil. If however the stencil is based on the smoothest set of nodes (i.e. only nodes on the left side of the discontinuity are used) the correct gradient is computed (figure 2.9b).

In figure 2.8, the grid resolution used for the signed distance function is too coarse to reproduce the sharp corner where the circles merge. Up sampling the signed distance functions using a spline before the interface is extracted helps to reveal the sharp corners (see figure 2.7) one would expect for two merging circles.

ENO schemes based on point values and TVD (total variation diminishing) Runge Kutta time discretisations for multiple space dimensions have been introduced by Shu & Osher (1988, 1989). In these schemes, the smoothest stencil is chosen at each node from a set of stencils which could be used to approximate the finite difference operator. Jiang & Shu (1996) and Liu et al. (1994) introduced weighted ENO (WENO) schemes as an improvement over the ENO schemes. They use a convex combination of all candidate stencils instead of just one as in the original ENO scheme.

The higher order approximation of the level set equation used in this work is based on a fifth order WENO discretisation in space together with a third order TVD Runge Kutta discretisation in time. This discretisation has previously been suggested by Jiang & Peng (2000). In general, there are many acceptable solvers for Hamilton-Jacobi equations that could be used to approximate the level set equation. One should also keep in mind that the field of solving hyperbolic conservation laws (e.g. Jiang & Tadmor, 1998; Ammar et al., 2006) and Hamilton-Jacobi equations (e.g. Serna & Qian, 2006; Lin & Liu, 2007) is vast and new techniques are continually under development. Comprehensive discussions of earlier work in the field of hyperbolic conservation laws have been given by LeVeque (1992) and Sod (1985).

In the scheme used here, which is based on a regular grid, space and time is discretised into a collection of grid points with a spatial spacing of Δx and Δy and a time step Δt. Every grid point is represented by a coordinate triplet (i, j, n) corresponding to the point $(i\Delta x, j\Delta y, n\Delta t)$. The value of the signed distance function ϕ at the node (i, j) of the grid at time $n\Delta t$ is given by $\phi^n_{i,j}$. The spatial derivatives are discretised using a fifth order WENO scheme in space. The forward Δ^+ and backward Δ^- finite difference approximations for the first derivative of the signed distance function in the x direction at the grid node (i, j) are given by

$$\Delta^{\pm}\phi_{x,i,j} = \frac{\pm(\phi_{i\pm1,j} - \phi_{i,j})}{\Delta x}. \tag{2.26}$$

The finite difference approximation to the second derivative in the x direction at the grid node (i, j) is given by

$$\Delta^{-}\Delta^{+}\phi_{x,i,j} = \frac{\phi_{i-1,j} - 2\phi_{i,j} + \phi_{i+1,j}}{\Delta x^2}. \tag{2.27}$$

Analogously one can define the finite difference operators in the y direction.

WENO schemes are basically centred schemes in regions where the solution is smooth. If there is a discontinuity inside the stencil, the WENO scheme can effectively choose the smoothest sub stencil for the approximation and therefore avoid undesirable oscillations (Jiang & Shu, 1996). The backward (left) finite difference in the x direction is given by the following fifth order WENO scheme (Jiang & Peng, 2000)

$$\begin{aligned}\Delta^{-}_{WENO}\phi_x = \frac{1}{12}(&-\Delta^{+}\phi_{x,i-2,j} + 7\Delta^{+}\phi_{x,i-1,j} + 7\Delta^{+}\phi_{x,i,j}\\ &-\Delta^{+}\phi_{x,i+1,j}) - \Delta x\Phi^{WENO}(\Delta^{-}\Delta^{+}\phi_{x,i-2,j},\\ &\Delta^{-}\Delta^{+}\phi_{x,i-1,j}, \Delta^{-}\Delta^{+}\phi_{x,i,j}\Delta^{-}\Delta^{+}\phi_{x,i+1,j}),\end{aligned} \tag{2.28}$$

where

$$\Phi^{WENO}(a, b, c, d) = \frac{1}{3}\omega_0(a - 2b + c) + \frac{1}{6}(\omega_2 - \frac{1}{2})(b - 2c + d) \tag{2.29}$$

with weights defined as

$$\omega_0 = \frac{\alpha_0}{\alpha_0 + \alpha_1 + \alpha_2}, \quad \omega_2 = \frac{\alpha_2}{\alpha_0 + \alpha_1 + \alpha_2},$$
$$\alpha_0 = \frac{1}{(\epsilon + \beta_0)^2}, \quad \alpha_1 = \frac{6}{(\epsilon + \beta_1)^2}, \quad \alpha_2 = \frac{3}{(\epsilon + \beta_2)^2};$$

$$\begin{aligned}\beta_0 &= 13(a - b)^2 + 3(a - 3b)^2,\\ \beta_1 &= 13(b - c)^2 + 3(b + c)^2,\\ \beta_2 &= 13(c - d)^2 + 3(3c - d)^2.\end{aligned}$$

Here ϵ is used to prevent the denominators from becoming zero. The solution is relatively insensitive to the values chosen for ϵ. Jiang & Peng (2000) use $\epsilon = 10^{-6}$ in their implementation of a fifth order WENO scheme. The fifth order WENO forward (right) finite difference in the x direction is given by

$$\begin{aligned}\Delta^{+}_{WENO}\phi_x = \frac{1}{12}(&-\Delta^{+}\phi_{x,i-2,j} + 7\Delta^{+}\phi_{x,i-1,j} + 7\Delta^{+}\phi_{x,i,j} \\ &-\Delta^{+}\phi_{x,i+1,j}) - \Delta x \Phi^{WENO}(\Delta^{-}\Delta^{+}\phi_{x,i+2,j}, \\ &\Delta^{-}\Delta^{+}\phi_{x,i+1,j}, \Delta^{-}\Delta^{+}\phi_{x,i,j}, \Delta^{-}\Delta^{+}\phi_{x,i-1,j}).\end{aligned} \tag{2.30}$$

By replacing the x direction with the y direction, expressions for $\Delta^{-}_{WENO}\phi_y$ and $\Delta^{+}_{WENO}\phi_y$ can be written. One can then replace the first order finite difference operators (e.g. $\Delta^{+}\phi_x$) for the spatial derivatives in (2.23) and (2.24) with the corresponding fifth order WENO finite difference operator (e.g. Δ^{+}_{WENO}) so that (2.22) becomes a scheme with fifth order accuracy in space.

So far the focus has been on the spatial discretisation. It is important to note that near discontinuities, the fifth order WENO scheme essentially steps back to a third order ENO scheme so that the finite difference operator is based on the smoothest stencil. Hence a third order accurate finite difference operator in time is sufficient to match the spatial WENO scheme. Here, the third order TVD Runge Kutta scheme given by Shu & Osher (1988) and Gottlieb & Shu (1998) is used. Thus, the finite difference approximation to the level set equation becomes

$$\phi^{1}_{i,j} = \phi^{n}_{i,j} - \Delta t[\max(F_{i,j},0)\nabla^{+}, \min(F_{i,j},0)\nabla^{-}], \tag{2.31}$$

$$\phi^{2}_{i,j} = \frac{3}{4}\phi^{n}_{i,j} + \frac{1}{4}\phi^{1}_{i,j} - \Delta t[\max(F_{i,j},0)\nabla^{+}, \min(F_{i,j},0)\nabla^{-}], \tag{2.32}$$

$$\phi^{n+1}_{i,j} = \frac{1}{3}\phi^{n}_{i,j} + \frac{2}{3}\phi^{2}_{i,j} - \frac{2}{3}\Delta t[\max(F_{i,j},0)\nabla^{+}, \min(F_{i,j},0)\nabla^{-}]. \tag{2.33}$$

2.2.3.4 Numerical realisation

A first order and higher order scheme for approximating the level set equation has been introduced. Given an interface which is to be tracked in time, one normally begins by initialising the signed distance function. When discretising the signed distance function and choosing the node separation and time step, it is important to remember that for each time step the numerical domain of dependence should contain the mathematical domain of dependence. In the one dimensional example (section 2.2.3.1), this is only the case if a point moving along the characteristics does not move more than Δx for a given time step Δt. This is also known as the CFL (Courant Friedrichs Lewy) condition (Courant et al., 1928). The stability of the scheme depends on a balance between the time step Δt, the space step Δx

and the speed F given by

$$\max(F)\Delta t \leq \Delta x, \tag{2.34}$$

where the maximum is taken over all values of F at all possible points within the domain, not simply those corresponding to the zero level set. Due to the CFL condition, increasing the grid resolution by a factor of two means that one must halve the time step.

The value of F must be known at all grid nodes of the signed distance function. This poses no problem if the speed is given by an analytical function. On the other hand, if the speed is only known on a coarser grid, then some form of interpolation has to be applied. In the following, a cubic B-spline approximation is used for the speed F. If F is given on a regular grid as $c_{i,j}$ its value at an arbitrary position (u, v) in the grid cell (i, j) is given by

$$B_{i,j}(u,v) = \sum_{l=-1}^{2} \sum_{m=-1}^{2} b_l b_m c_{i+l,j+m}, \tag{2.35}$$

where $u(0 \leq u \leq 1)$ and $v(0 \leq v \leq 1)$ are expressed as a fraction of the grid spacing. The weighting factors b_l, b_m are the uniform cubic B-spline functions (Bartels et al., 1987). When approximating the level set equation having a smooth speed function F is an advantage from a stability point of view. A cubic B-spline approximation has the property that the resulting field has continuous first and second derivatives (see appendix B), yet is locally controlled.

The computational domain is finite, so the finite difference stencils used to approximate the spatial derivatives may extend outside of the computational domain when working near a boundary. This means that ghost nodes have to be added to the computational grid. Two types of boundary conditions are used.

1. Periodic boundary conditions: For each dimension, values from one end of the grid are copied across the grid to the ghost nodes at the other end of the grid and vice versa.

2. Extrapolating boundary conditions. The values of the signed distance functions are for each dimension linearly extrapolated away from zero into the ghost nodes.

Periodic boundary conditions should be used whenever they can be justified. Otherwise, the extrapolating boundary conditions are quite useful in level set computations. By extrapolating away from zero the ghost node will never falsely imply the existence of a ghost interface, as the value of the signed distance function cannot become 0 for a ghost node.

Using all the concepts introduced so far, it is now possible to propagate an interface by updating the signed distance function for the whole computational domain. This is the so-called global level set method (e.g. Sethian, 1999; Osher & Fedkiw, 2003). However such an approach has two disadvantages. Firstly, only a few nodes of the underlying grid are actually near the interface and are essential to describing its position. Nodes distant from the zero level set still get updated but they tend to have little to no influence on the zero level set. Hence it can be inefficient to update all nodes of the underlying grid for each time step. Secondly, despite the stability preserving finite difference solvers, a signed distance function might develop kinks and could become very flat ($(|\nabla\phi < 1|)$) or steep ($(|\nabla\phi > 1|)$) near the interface. It therefore would be desirable to regularise the signed distance function from time to time so that the value of the gradient of the signed distance function is again equal to one for most nodes.

2.2.4 Improved algorithms for interface propagation

A global level set method can be improved using the following concepts:

- Reinitialisation: During the propagation process a signed distance function may become less well behaved i.e. $|\nabla\phi| \neq 1$. As this influences the accuracy of the scheme a reinitialisation of the signed distance function can be performed so that it is again well behaved for most of the grid nodes i.e. $|\nabla\phi| = 1$. This will generally increase the accuracy, especially for complex interface motion.

- Narrow band: In general only the zero level set of the signed distance function is of interest. One way of increasing the efficiency of the method involves updating the signed distance function in a narrow band around the zero level set. As the interface moves, the narrow band must be moved with it.

- Adaptive mesh refinement: A signed distance function can only capture features of an interface which extend over several grid cells. Hence increasing the resolution where the zero level set exhibits high curvature would be desirable. On the other hand a high grid resolution away from the zero level set is not required. Therefore an efficient level set method can be constructed using an adaptive mesh refinement strategy such as the one

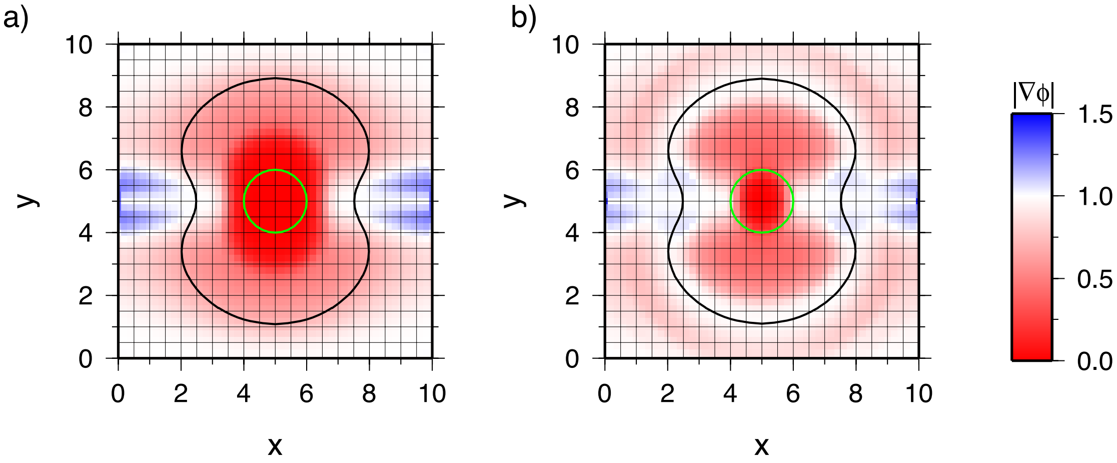


Figure 2.10: A circle (green line) has been expanded with a speed F in its outward normal direction given by $F(x, y) = 1.0/(1.0 + \exp(-(y - 5.0)^2)$. (a) Gradient of the signed distance function and zero level set (black line) with no reintialisation. (b) Same as (a) except now with reintialisation after each time step.

presented by Berger & Colella (1989) for solving hyperbolic conservation laws in two dimensions.

Recent implementations of the level set method tend to combine all three concepts (e.g. Losasso et al., 2006). Here, however, the focus will be on a narrow band level set method which requires that the signed distance function can be reinitialised at any given node. Hence, the concept of reinitialisation is introduced first.

2.2.4.1 Reinitialisation

For numerical accuracy the signed distance function has to be well behaved for most nodes of the grid. This means that except for isolated grid nodes the value of its gradient should be equal to one i.e.

$$|\nabla \phi| = 1. \tag{2.36}$$

In figure 2.10a, a circle has been expanded in its local normal direction with a speed given by $F(x, y) = 1.0/(1.0 + \exp(-(y - 5.0)^2)$. Instead of the value of the signed distance function, the value of its gradient is plotted. Clearly, it is no longer equal to one near the interface. Therefore a procedure is needed to reset ϕ so that it is well behaved in the neighbourhood of the front. Chen & Giga (1991) and Evans & Spruck (1991) showed that functions other than a signed distance function can be used in the level set method provided that initially the interface coincides with the zero level set. From this point of view it is possible to modify a signed distance function, and as long as the zero level set is not moved, the accuracy should not be affected.

One approach would be to extract the interface with some interpolation technique and then to calculate a new signed distance function for this interface (Merriman et al., 1994). Such an approach would not move the interface within the numerical accuracy of the interpolation scheme. On the other hand, this is a rather inefficient approach and there is a chance that some spurious irregularity could be introduced, making gradient based properties of the interface nodes poorly behaved. Adalsteinsson & Sethian (1995) used the fast marching method to reconstruct the signed distance function around an interface successfully. Nodes which define a cell through which the interface passes are then used as a set of initial grid points for the fast marching method. A new signed distance function is constructed using the fast marching method away from the interface with a speed of one. The negative and positive part of the signed distance function are constructed separately. The fast marching method is unconditionally stable and efficient, so the resulting signed distance function is likely to be free from irregularities.

A more elegant way to rebuild the signed distance function has been presented by Sussman et al. (1994). They solve the so called reinitialisation equation, which is given by the following Hamilton-Jacobi type equation,

$$\phi_t + S(\phi)(1 - |\nabla\phi| = 0, \tag{2.37}$$

where $S(\phi)$ is the sign function which returns $+1$ if the signed distance function ϕ is positive and -1 if ϕ is negative. A higher order finite difference scheme has been introduced earlier for the level set equation, which is also a Hamilton-Jacobi type equation. Hence the same finite difference operators can be used to approximate (2.37). In this case ϕ is propagated with a speed of one away from the interface along the characteristics, which are normal to the interface. Eventually ϕ converges to a well behaved signed distance function ($|\nabla\phi| = 1$) in the neighbourhood of the interface.

The issue with applying (2.37) to a signed distance function is that the crudeness of the sign function $S(\phi)$ might cause considerable motion of the zero level set during reinitialisation. Sussman et al. (1994) therefore approximate $S(\phi)$ with a smooth function

$$S(\phi) = \frac{\phi}{\sqrt{\phi^2 + \epsilon^2}}, \tag{2.38}$$

where ϵ is set to Δx, the grid spacing. The smooth function (2.38) causes $S(\phi)$ to be close to zero near the interface. The signed distance function is therefore only moved slowly near the interface and the risk of accidentally moving the zero level set is reduced.

One iteration of the reinitialisation scheme per time step proved sufficient for their example of a rising air bubble in water (Sussman et al., 1994). For this fluid air interaction problem, they show that reinitialisation is critical for the conservation of mass and preserving the correct shape of the air bubble. In solving the reinitialisation equation they choose a time step of $\frac{1}{10}\Delta t$, where Δt is the time step used for approximating the level set equation.

This approach works well when the signed distance function ϕ is neither too flat ($|\nabla\phi| << 1$) nor too steep ($\nabla\phi >> 1$) near the interface. When the signed distance function is too flat $|\phi|$ approaches zero and (2.38) becomes small and the resulting speed for the propagation of ϕ along the characteristics also diminishes. Several steps are therefore necessary to reconstruct ϕ as a well behaved signed distance function near the interface. If, on the other hand, $|\phi|$ becomes large near the interface this approach might change the sign of ϕ and hence move the interface.

Peng et al. (1999) therefore suggest a numerical approximation for the sign function $S(\phi)$ which depends on the local slope of ϕ and hence guarantees that the sign of ϕ does not change:

$$S(\phi) = \frac{\phi}{\sqrt{\phi^2 + |\nabla\phi|^2\Delta x^2}}, \tag{2.39}$$

where $\nabla\phi$ is estimated using the spatial finite difference operators introduced before and Δx is the grid spacing. If, in a higher dimensional space, the grid spacing varies according to direction, then the average grid spacing could be used for Δx. It is however advisable to use the same grid spacing in all directions, as this increases the stability of the numerical solvers (e.g. Jiang & Peng, 2000). This formulation of the sign function $S(\phi)$ solves the problem of changing the sign of ϕ and accidentally moving the interface across the cell boundary if $|\nabla\phi| >> 1$. It will also speed up the convergence when $|\nabla\phi| << 1$.

For a circle expanding under a smooth speed function F in its normal direction there is no obvious change in the position and shape of the interface with the application of reinitialisation (see figure 2.10). However, in more complex situations the effects of reintialisation will be more evident. In practice it is often sufficient to perform one step of the reinitialisation procedure every few time steps in order to keep a signed distance function well behaved (e.g. Sussman et al., 1994; Osher et al., 2002).

2.2.4.2 Narrow band approach

Updating the signed distance function in the region of interest around the zero level set leads to narrow band level set methods (e.g. Peng et al., 1999; Sethian, 1999; Osher & Fedkiw, 2003). There are two reasons for developing a narrow band approach if the speed F is given on the whole domain.

- Minimise number of nodes: Considerable savings in computational effort can be achieved by only considering nodes in the neighbourhood of the zero level set; i.e. in a narrow band. As the front propagates grid nodes are added and removed from the narrow band.
- Maximise time step: The global level set method requires that the CFL condition is satisfied with regard to the maximum speed over the whole domain. In a narrow band level set method, the CFL condition only needs to be satisfied with respect to the maximum speed in the narrow band. If the speed of the front changes substantially as the front evolves, the CFL condition for the whole domain is much more stringent than the local CFL condition within the narrow band.

The narrow band level set method was introduced by Chopp (1993) for the computation of surfaces of least area constrained by a given boundary (i.e. surface minimisation). Soap films within a wire boundary are an example of surfaces of least area arising naturally. A narrow band approach decreases the computation time, but on the other hand requires the calculation of the signed distance function for points being added to the narrow band as the front evolves.

Once a signed distance function has been initialised for a given interface grid, nodes belonging to the narrow band have to be identified. The narrow band is then defined by the set of grid nodes for which the value of ϕ is smaller than a threshold β. The narrow band defined by β will in the following be called the inner band. The minimum width of the inner band depends on the width of the stencil of the scheme used to approximate the spatial derivatives. For a fifth order WENO scheme, at least three grid nodes should be used on either side of the interface (Peng et al., 1999), and hence $\beta = 3\Delta x$. If all nodes of the inner band are to be updated, information from outside the inner band is required by the finite difference operator. Therefore, an outer band is defined by all the grid nodes for which $\beta < |\phi| < \gamma$ where $\gamma > \beta$. For a fifth order WENO scheme the outer band is given by $\gamma = 6\Delta x$. If a node is in the outer band it contains a valid value for the signed distance function, but its value is not updated using the approximation to the level set equation. Nodes which belong to the outer band are updated using a reinitialisation procedure such as the one based on the fast marching method or on the reinitialisation equation. As the front propagates, points are added and removed from the inner and outer band. The complete narrow band essentially consists of an inner and an outer band.

Having defined the inner band and outer band the steps required to update the signed distance function and hence to propagate the interface are outlined below (see also figure 2.11).

1. In the first step nodes belonging to the inner band are updated using the approximation to the level set equation introduced before.

2. Grid nodes neighbouring the narrow band are added to the outer band.

3. A reinitialisation of the signed distance function is performed for all nodes belonging to the inner and outer bands.

4. Nodes are added and removed from the inner and outer bands depending on the value of the signed distance function.

In step three, the spatial finite difference operator might require information from nodes beyond the limits of the outer band. As these nodes are distant from the interface, and hence do not strongly influence the position of the interface, one could use a smaller stencil or extrapolate the values of the signed distance functions. An alternative to approximating the reinitialisation equation in step 3 is to use the fast marching method for the reconstruction of the signed distance function, which has the advantage that it only requires nodes adjacent to the interface, for which the signed distance function is known (Adalsteinsson & Sethian, 1995).

In figure 2.12 an interface begins as a straight line and propagates to the right in the outward normal direction with a speed $F(x, y) = 1.0/(1.0 + \exp(-(y - 5.0)^2))$.

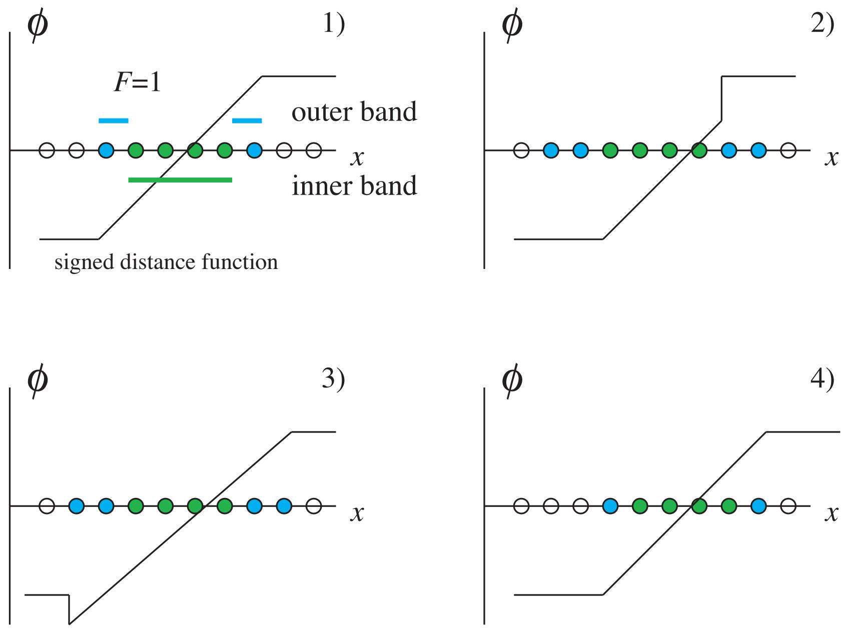


Figure 2.11: The four steps required for updating a signed distance function in a narrow band. The narrow band consists of an inner and outer band. The front is moving to the right with a speed $F = 1$. Refer to text for more details.

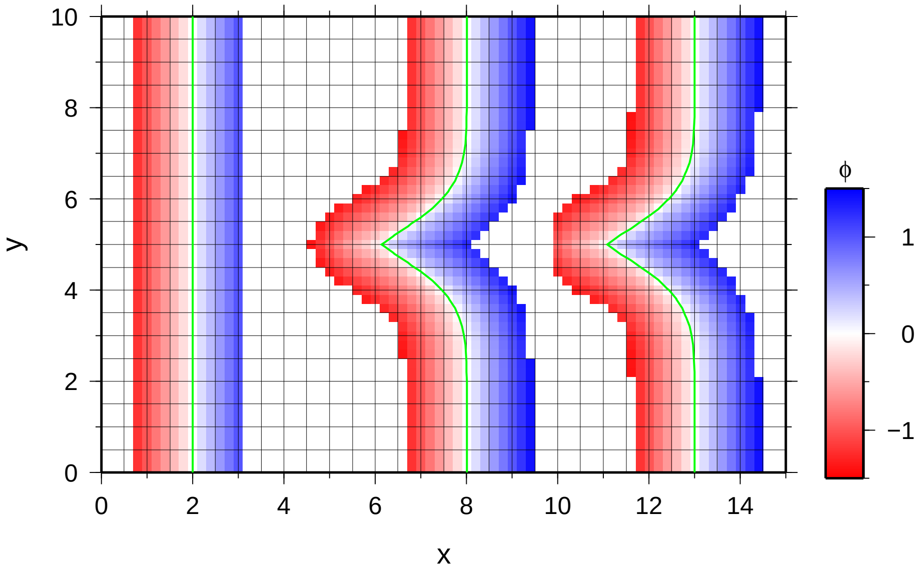


Figure 2.12: An interface (green line) propagating to the right with a speed $F(x,y) = 1.0/(1.0+\exp(-(y-5.0)^2))$. The position of the interface and the corresponding signed distance function in the narrow band is plotted at three different times.

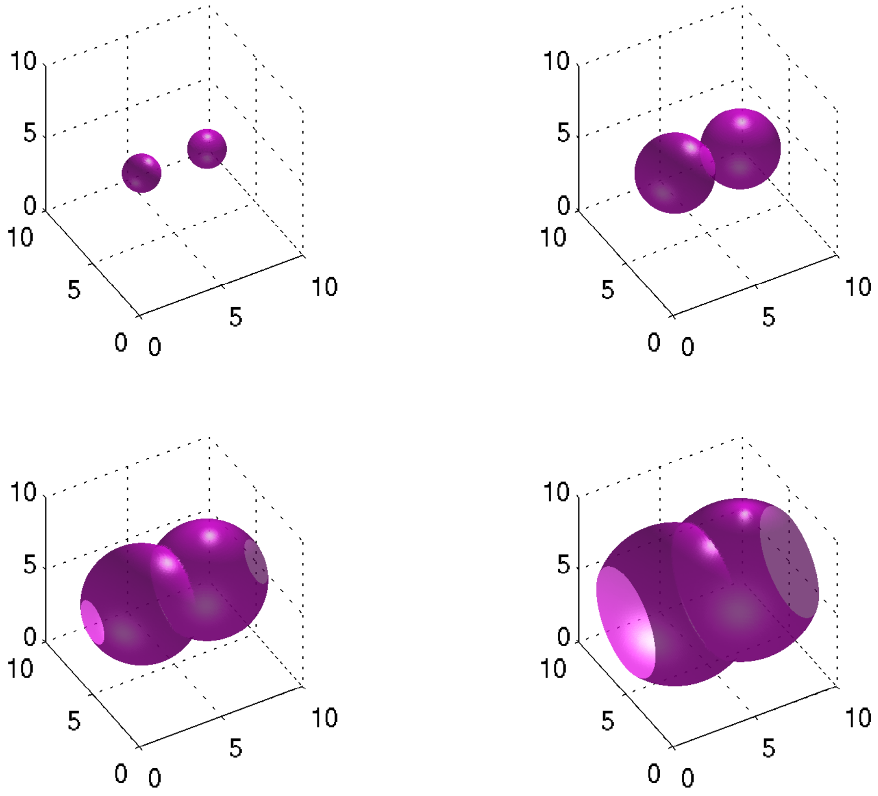


Figure 2.13: Two spheres propagating with a speed $F = 1$ in their outward normal direction. The spheres eventually merge and begin to leave the computational domain.

The narrow band is plotted at three different times. In this example, the signed distance function has been reinitialised after each time step using the fast marching method. As one can see, the interface is adequately described by the grid nodes in its neighbourhood.

2.2.5 Three dimensions

As discussed previously, schemes for higher dimensional Hamilton-Jacobi equations for symmetric Hamiltonians can be built by simply replicating each space variable. This allows the approximation to the level set equation and reinitialisation equation to be formulated for the evolution of an $m-1$ dimensional manifold in an m dimensional space. In three dimensions, the level set method describes the evolution of a surface. Figure 2.13 shows two spheres which propagate in their outward normal direction with a speed $F = 1$ and eventually merge. This is sim-

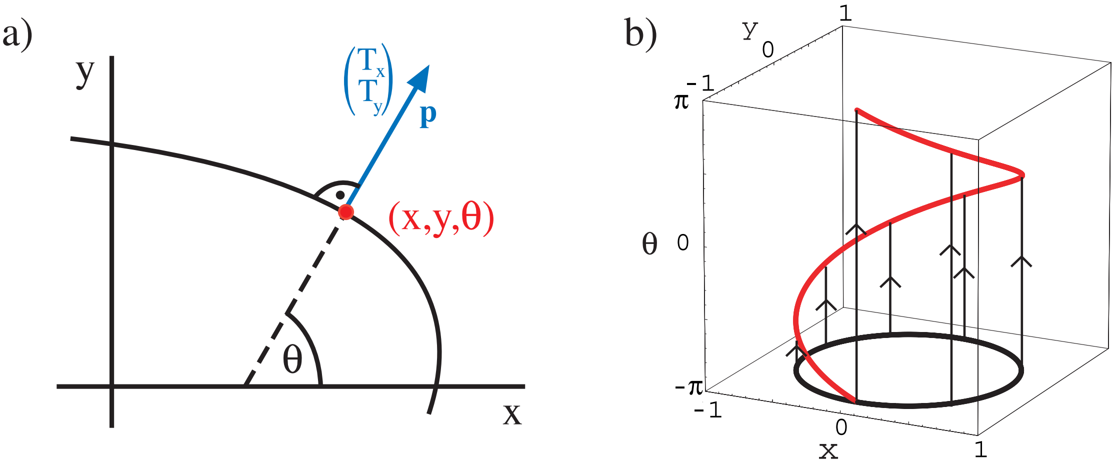


Figure 2.14: Reduced phase space representation of a two dimensional wavefront. (a) Ray trajectory from the wavefront supplies third dimension θ. (b) Wavefront in normal space (black line) and corresponding bicharacteristic strip (red line) in reduced phase space.

ilar to the two dimensional problem with the two expanding circles in figure 2.7. The grid consists of $101 \times 101 \times 101$ nodes and the time step is given by $\Delta t = 0.1\,\text{s}$. Increasing the grid resolution by a factor of two in the two dimensional case will lead to an increase in computation time by a factor of eight. In three dimensions the corresponding increase will be sixteen fold.

Figures 2.13 and 2.7 emphasise that interfaces approximated using the level set method can only merge or break but not intersect each other. A seismic wavefront, however, can become self intersecting if later arrivals are generated; for example, when a swallowtail pattern develops in the presence of a low velocity anomaly (see figure 1.1). Hence, the level set method cannot be used directly to describe the wavefront. Osher et al. (2002) therefore unfold the wavefront from normal space into reduced phase space, where the wavefront does not self-intersect.

2.3 Reduced phase space

The state of a particle on the wavefront is characterised by its position and motion. Calculating the position of a point on the wavefront at a later moment in time requires some information about the motion. In kinematic ray tracing this information is normally carried along using the slowness vector (e.g. Vinje et al., 1993; Červený, 2001).

Wavefront tracking in phase space using a Lagrangian framework has been done previously by Lambaré et al. (1996) and Lucio et al. (1996) using the Hamiltonian formulation of ray theory. For a wavefront in two dimensional real space (or normal space) a four dimensional phase space can be constructed where the four coordinates are the two components of the position vector and the slowness vector. The characteristics of the eikonal equation (see (2.2)) in phase space are given by (e.g. Červený, 2001; Chapman, 2004)

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = c\frac{\mathbf{p}}{|\mathbf{p}|}, \tag{2.40}$$

$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = -|\mathbf{p}|\nabla c, \tag{2.41}$$

where $\mathbf{p} = \nabla T$ is the slowness vector, $\mathbf{x}$ the position vector in real space, c the speed and t the time. This is the Hamiltonian formulation of ray theory, where the rays are the bicharacteristics in $(\mathbf{x}, \nabla T)$ (or just characteristics in $\mathbf{x}$) and the wavefront is generally known as the bicharacteristic strip (e.g. Chapman, 1985; Osher et al., 2002). It defines the direction and location of the wavefront.

Instead of using four dimensional phase space it is also possible to define a reduced phase space which is three dimensional for a wavefront in two dimensional real space. The third component is the direction of the slowness vector defined by the angle the slowness vector forms with the x axis (see figure 2.14). The two components of the slowness vector describing the motion of a node on the wavefront T_x and T_y are essentially combined into θ, the direction of the local wavefront normal or slowness vector, using the relationship $\tan\theta = (T_y/T_x)$.

To transform a wavefront from two dimensional real space into three dimensional reduced phase space, one can calculate the direction θ of the local wavefront normal and use it as the third coordinate. Provided the wavefront has curvature, different points along the wavefront will have different θ values and hence plot at different positions in reduced phase space. Thus the bicharacteristic strip will not self-intersect. The other advantage of reduced phase space is that wavefronts containing sharp corners will be described by a locally smooth bicharacteristic strip. Instead of tracking a self-intersecting wavefront with sharp corners in two dimensional real space, the locally smooth and non self-intersecting bicharacteristic strip (Osher et al., 2002) can be tracked in reduced three dimensional phase space (see figure 2.15). Whenever the position of the wavefront is required, the bicharacteristic strip is mapped back into normal space.

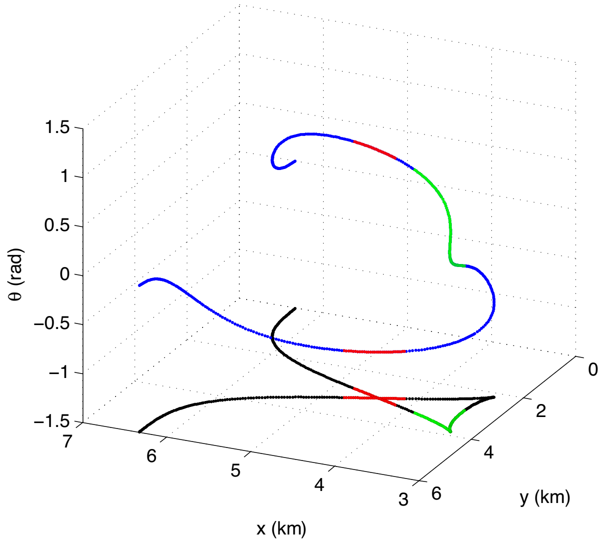


Figure 2.15: Swallowtail pattern of a wavefront. The sharp corners in normal space (green segments) are given by a smooth representation in reduced phase space. The intersecting segments in normal space (red lines) do not intersect each other in reduced phase space.

2.4 Propagating the bicharacteristic strip in an Eulerian framework

The level set method can only describe the evolution of a surface in three dimensions. It cannot be directly used to evolve a line in three dimensions. Osher et al. (2002) therefore describe the bicharacteristic strip as the intersection of two surfaces (see figure 2.16), which in turn are represented as the zero level sets of two signed distance functions ϕ and ψ. The bicharacteristic strip is then given by the set of points where $\phi = \psi = 0$.

The two surfaces are described using the signed distance functions ϕ and ψ, and are evolved separately by solving the following set of equations;

$$\begin{aligned} \phi_t + \mathbf{v}\nabla\phi &= 0, \\ \psi_t + \mathbf{v}\nabla\psi &= 0, \end{aligned} \tag{2.42}$$

where $\mathbf{v}$ is the velocity field in reduced phase space. A particle on the bicharacteristic strip moves along the bicharacteristics in reduced phase space, which are given by the characteristics of the eikonal equation. Thus the components of the

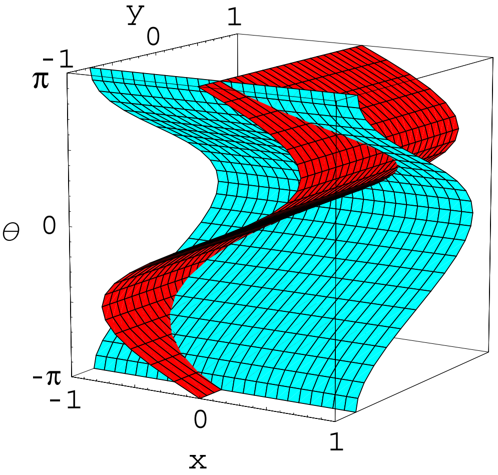


Figure 2.16: In the Eulerian scheme, the bicharacteristic strip is represented as the intersection of two implicitly defined surfaces. The bicharacteristic strip illustrated here is the same as the one shown in figure 2.14.

velocity vector **v** for this particle are (Osher et al., 2002)

$$\begin{aligned}\frac{\mathrm{d}x}{\mathrm{d}t} &= c\cos\theta,\\ \frac{\mathrm{d}y}{\mathrm{d}t} &= c\sin\theta,\\ \frac{\mathrm{d}\theta}{\mathrm{d}t} &= c_x\sin\theta - c_y\cos\theta, \end{aligned} \tag{2.43}$$

where c is the speed function given in real or normal space, and c_x and c_y are its derivatives in the x and y direction. Equation 2.43 is the initial value formulation of the kinematic ray tracing equation in isotropic media, which describes the characteristics of the eikonal equation (e.g. Červený, 1987, 2001). A similar set of equations can be derived for the propagation of rays in anisotropic media. Qian et al. (2003) present a scheme for two dimensional anisotropic wavefront tracking by replacing (2.43) with a set of equations for ray tracing in anisotropic media. Their grid based scheme is an alternative to ray tracing, which is commonly used in anisotropic media (e.g. Kendall & Thomson, 1989; Červený, 2001).

In the scheme presented here the speed c is defined at the nodes of a grid, and cubic B-spline approximation is used to compute the speed c and it derivatives c_x and c_y at an arbitrary position.

Given a point source in real space at the position (x_s, y_s) the bicharacteristic strip becomes a straight line $(x_s, y_s, -\pi \leq \theta \leq +\pi)$. This line is defined as the intersection of two surfaces which are perpendicular to each other and given by the zero level sets of the two signed distance functions ϕ and ψ. The signed distance functions for the two planes $x = x_s$ and $y = y_s$ are initialised using the following set of equations;

$$\begin{aligned} \phi(x, y, \theta) &= x_s - x, \\ \psi(x, y, \theta) &= y_s - y. \end{aligned} \tag{2.44}$$

Instead of tracking the wavefront in real space the two signed distance functions are updated using the level set equation (2.8). Essentially, two three dimensional problems need to be solved in order to find the solution to a two dimensional problem.

To plot the position of the wavefront at a given moment in time, the bicharacteristic strip needs to be extracted from the two signed distance functions ϕ and ψ. In order to find the bicharacteristic strip on the grid, cubes of the grid where both signed distance functions change their sign are located. Each cube is defined by eight nodes and may be divided into six space filling tetrahedra. If both signed distance functions change their sign in a given tetrahedron, the intersection between the two zero level sets (i.e. the bicharacteristic strip) must pass through this tetrahedron. Therefore the bicharacteristic strip must enter the tetrahedron through one face and leave it through another face. The intersection points of the two signed distance functions on the two faces of the tetrahedron can then be computed. The union of all these intersection points provides the bicharacteristic strip, which then can be mapped back into real space in order to plot the wavefront. This algorithm has previously been suggested by Burchard et al. (2001) and is essentially based on a simplification of the marching cube method described by Lorensen & Harvey (1987).

2.4.1 Arrival time extraction

In seismological applications the arrival time is usually more useful than the position of the wavefront at a given moment in time. One way to extract an arrival time is to store, for each time step Δt and each receiver, the square of the distance to the closest point on the wavefront. The recorded distance forms a local minimum if a wavefront has passed a receiver. Fitting a parabola through the three points defining the local minimum allows the interpolation of an arrival time (see

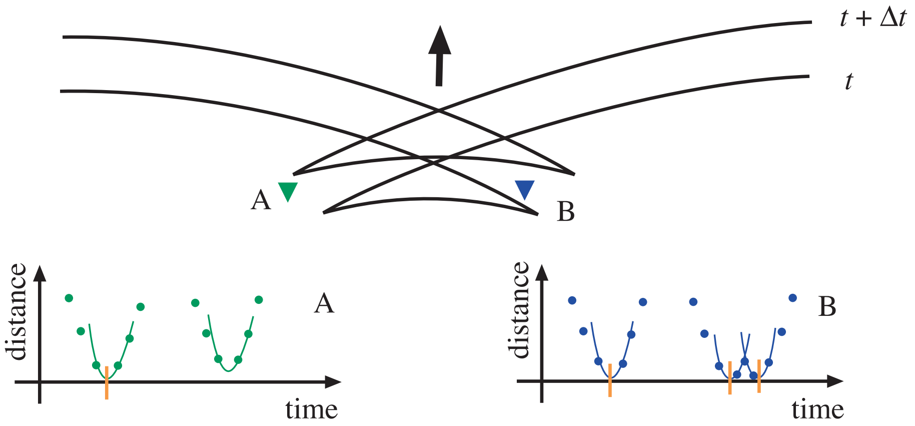


Figure 2.17: Wavefronts exhibiting a swallowtail pattern. For both receivers A and B, the discrete squared distance between the receiver and the closest point on the wavefront is plotted. Where three points form a local minimum a parabola is fitted through them. It is unclear whether or not the left endpoint of the wavefront has intersected receiver A, while receiver B has registered three clear arrivals.

figure 2.17).

The advantage of this approach is that there is no need to calculate the bicharacteristic strip explicitly. In phase space, a receiver becomes a line and therefore the squared distance to the closest point on the bicharacteristic strip for a receiver at position $\mathbf{x}$ is given by $\min(\phi(\mathbf{x})^2 + \psi(\mathbf{x})^2)$. Only arrivals of wavefronts which are at least Δt apart can be detected. If the temporal separation between two wavefronts is smaller than Δt, they are registered as one arrival. The squared distance to the closest point on the bicharacteristic strip for a receiver at position $\mathbf{x}$ is based on a discretised representation of the two signed distance functions. Hence, the distance to the closest point on the bicharacteristic strip might not be equal to zero even if the strip intersects the line representing the receiver. Therefore, a threshold has to be used in order to distinguish between true arrivals and the passage of endpoints (e.g. receiver A in figure 2.17) close to a receiver.

2.5 Examples

A series of examples are presented to investigate the capability of the Eulerian scheme. The first example uses a point source in a constant velocity model, which allows the influence grid resolution has on the error pattern to be evaluated. The scheme is then used to track the evolution of a plane wave that impinges on a wave guide structure. Finally, an attempt is made to calculate travel times using

grid size	number of time steps	time step (s)	computation time (s)
$51 \times 51 \times 51$	100	0.1	128
$101 \times 101 \times 101$	200	0.05	1630

Table 2.1: Table showing the computation time for the two configurations used for the test with a constant velocity model.

a velocity model for the subduction zone in the Tonga region. The method has been implemented under GNU/Linux in Fortran and all computation times are given for a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory.

For all examples the global level set method is used in preference to the narrow band level set method, which proved to be difficult to implement in three dimensions. Ultimately a global level set method is sufficient to investigate the feasibility of an Eulerian approach; the additional effort required to implement an efficient narrow band method is only worthwhile if the underlying level set scheme proves itself to be at least comparable to Lagrangian wavefront tracking.

2.5.1 Constant velocity

A point source is placed in a medium with a constant velocity of 1 km/s. While this is a rather simple model, it allows the error pattern to be examined and provides an estimate of the maximum achievable accuracy. Travel times are calculated using the two different configurations given in table 2.1. As expected, increasing the grid resolution by a factor of two leads to an increase in computation time by a factor of about sixteen.

In figure 2.18 the difference between the analytical and numerical solution is given in percent for the the two configurations. The source is located in the centre. As one would expect, increasing the grid resolution reduces the overall error. For both configurations the error is smaller than the time step used for the update of the signed distance function. There are two potential sources of error. (1) The accuracy of the numerical procedures used for updating the signed distance functions is limited; (2) as outlined before, the accuracy of the procedure used for the extraction of travel times is limited by the grid resolution.

The RMS error when $51 \times 51 \times 51$ nodes are used is 0.087 %, which compares to 0.022 % when $101 \times 101 \times 101$ nodes are used. Halving the grid spacing decreases the RMS error by a factor of four. Decreasing the grid spacing by a factor of two also requires a halving of the time step, due to the CFL condition. One would

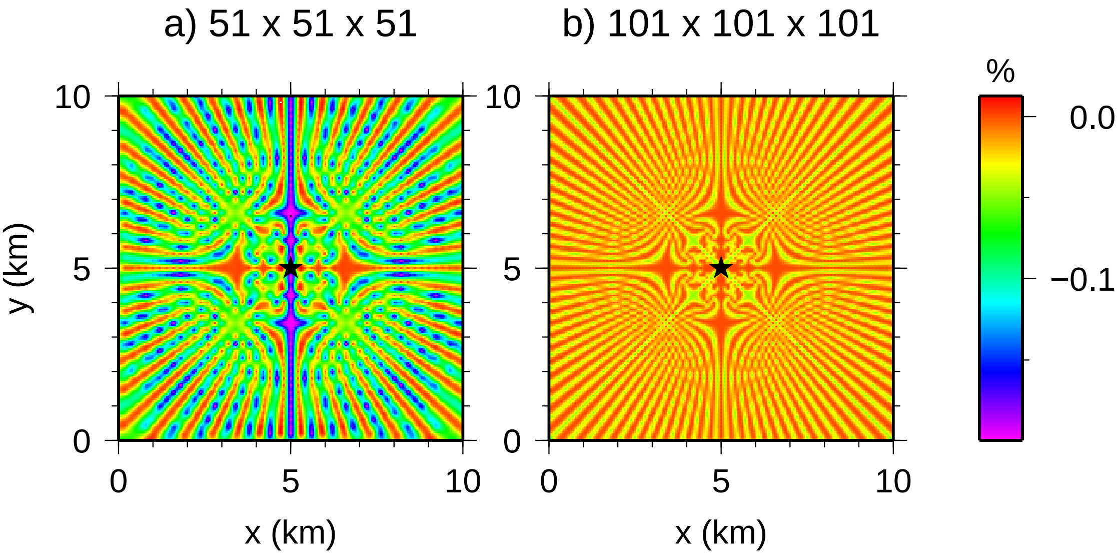


Figure 2.18: Relative error between the analytical and numerical solution for the two different grid resolutions given in table 2.1. The source location is marked by the black star. The overall RMS error in (a) is 0.087 % and in (b) 0.022 %.

therefore expect an increase in accuracy by a factor of four.

The error patterns in figure 2.18 show some symmetry and exhibit a strong small scale spatial variability. This is due to the two signed distance functions being defined on a grid. The calculated minimum squared distance from the receiver to the bicharacteristic strip can therefore be larger than the true squared distance due to a discretisation error. As the error of the calculated minimum squared distance is not constant, the accuracy of the extracted travel times is variable, but always smaller than the time step.

Taking computation time into consideration (see table 2.1), this simple example indicates that the Eulerian scheme may be a computationally expensive way to predict multi valued travel times.

2.5.2 Wave guide structure

For a plane wave in three dimensional reduced phase space the bicharacteristic strip is a line with constant θ, because all nodes on a plane wave move in the same direction. In this example (see figure 2.19) a plane wave begins near the rear (i.e. $x = 0.5$) of a wave guide model with a ±33 % variation in wave speed. The

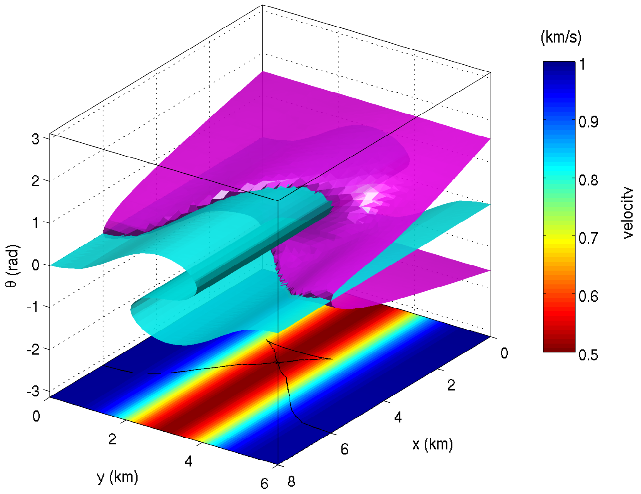


Figure 2.19: Zero level set of the signed distance functions ϕ (magenta) and ψ (cyan). The velocity field is plotted at the base of the cube. The wavefront (black line) is plotted for a time of 6 s. The wavefront is jagged because the grid used for the two signed distance functions is too coarse. The grid size is 51 × 31 × 51 and the computation time is 44.9 s with a time step of 0.1 s.

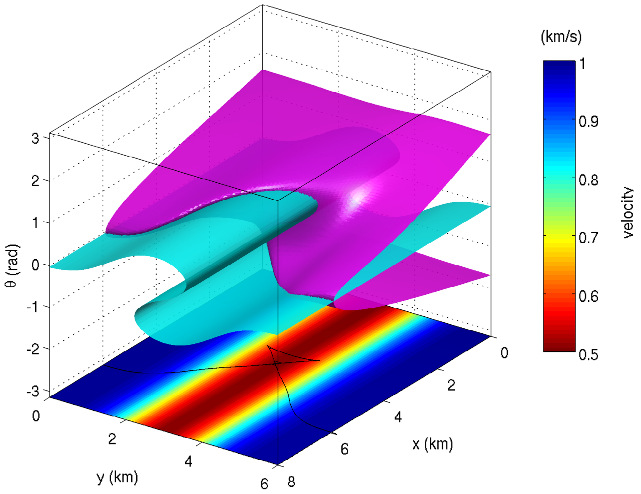


Figure 2.20: Zero level set of the signed distance functions ϕ (magenta) and ψ (cyan). The grid size for the computation is the same as in figure 2.19. The surfaces and the wavefront were extracted after having up-sampled the signed distance function by a factor of three using a cubic B-spline approximation. The wavefront (black line) is plotted for a time of 6 s. The wavefront exhibits some large scale undulation.

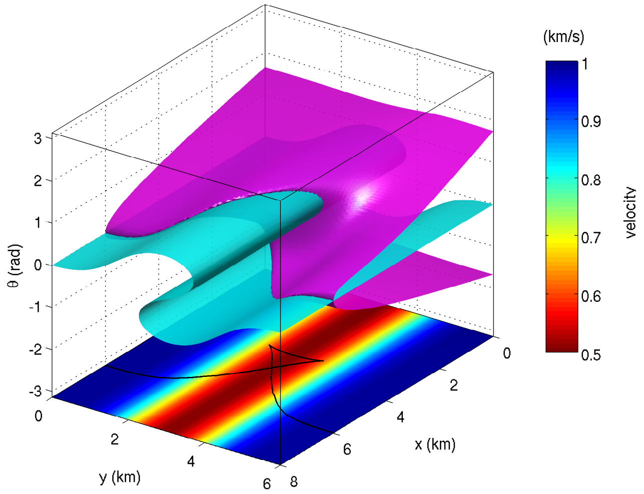


Figure 2.21: Zero level set of the signed distance functions ϕ (magenta) and ψ (cyan). The wavefront (black line) is plotted for a time of 6 s. The grid size is 151 × 91 × 151 which is sufficient for resolving the two interfaces and the resulting wavefront is smooth. The computation time is 3564.2 s and time step is 0.033 s.

equations for the initialisation of the two signed distance functions are given by.

$$\begin{aligned}\phi(x,y,\theta) &= x - x_s,\\ \psi(x,y,\theta) &= \theta_i,\end{aligned} \tag{2.45}$$

where the position of the initial plane wave is given by x_s and its direction by θ_i.

In figure 2.19 the plane wave propagates in the positive x direction towards the front of the model. The wave speed varies smoothly between 0.5 and 1.0 km/s and is given by the following function $c(x,y) = 1/(1+\exp(-(y-3)^2)$. The slow region in the centre of the model means that the wavefront starts to develop a swallowtail pattern. As Figure 2.19 shows, there are significant differences in orientation between the patches representing the two surfaces in reduced phase space. Any algorithm for the extraction of iso-surfaces based on the marching cube principle cannot provide a smooth representation unless there is some higher order interpolation taking place (Lorensen & Harvey, 1987).

Therefore, in figure 2.20, the two signed distance functions have been upsampled by a factor of three using cubic B-spline approximation before the bicharacteristic strip and hence the wavefront are extracted. The resulting surfaces appear smooth. The wavefront also looks smoother, but the shape still differs from the wavefront obtained when using a higher grid resolution during the computation (see figure 2.21). A higher order approximation provides a smoother representation, but this does not help, as it is the resolution of the underlying grid used for the update of the signed distance function that limits the complexity of the two surfaces, and hence the accuracy of the representation. As time evolves, parts of the surface represented by the signed distance function start to overturn within a single grid cell. It is not possible to represent such behaviour using the zero level set of a signed distance function defined on the nodes of a grid. Consequently, it is no longer possible to reconstruct the surface properly and information is lost.

In figure 2.21 the same wavefront has been calculated but the grid resolution has been increased by a factor of three. The signed distance functions resolve the two surfaces adequately, so the resulting wavefront is smooth. However, if this wavefront continues to propagate, the grid resolution will at some future moment in time no longer be high enough to fully describe the behaviour of the two surfaces. One has also to keep in mind that due to the increase in the grid resolution by a factor three, the memory requirements have increased by a factor of 27, and the computation time by a factor of 81.

2.5.3 Subduction zone

In a subduction zone, multiple arrivals from local earthquakes can appear due to the high velocity anomaly caused by the subduction of the cold oceanic lithosphere. For this test, a two dimensional P-wave velocity model of a subducting slab in the Tonga region (Conder & Wiens, 2006) is used[1] (see figure 2.22). The plane of the model is perpendicular to the strike of the subducting slab, and extends for 1400 km in the east west direction. The original tomographic P-wave model is described using bilinear interpolation on a rectangular grid of 57×29 nodes spaced 25 km horizontally and vertically. A cubic B-spline approximation will create a smoother velocity model than a bilinear interpolation using the same grid of velocity nodes. Therefore, to mitigate the effects of extra smoothing imposed by the cubic B-spline approximation, the velocity model is up-sampled before computing the travel times. The P-wave velocity model is then given by 113×57 nodes spaced 12.5 km horizontally and vertically.

In the first test, the signed distance functions are defined on a grid consisting of $113 \times 57 \times 61$ nodes. The grid is scaled via $x \to x/100$ and $y \to y/100$ so that

[1] The model used here and in section 3.2.3 was supplied by Wiens (2005, personal communication) and differs slightly from that which appears in the referenced paper.

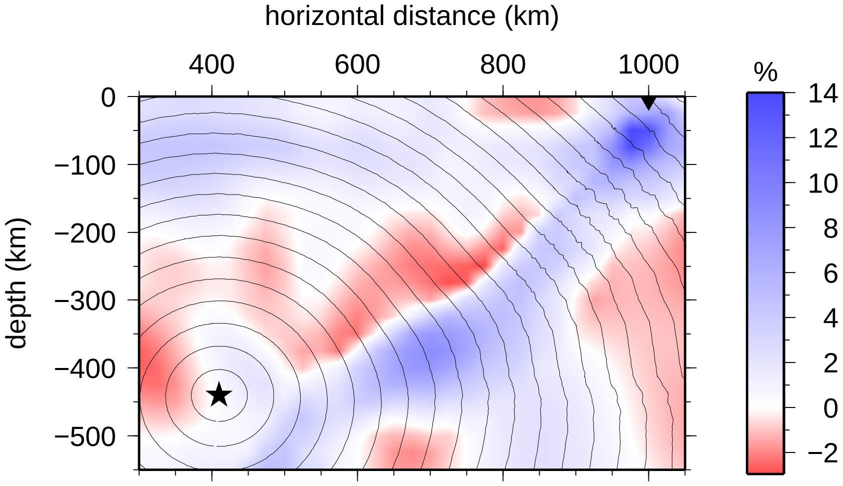


Figure 2.22: Wavefronts computed for a source west of the subducting slab. The grid size for the signed distance function is $113 \times 57 \times 61$. In this and the following figure, the wave speed is plotted as a perturbation with respect to a locally derived one dimensional model (Conder & Wiens, 2006).

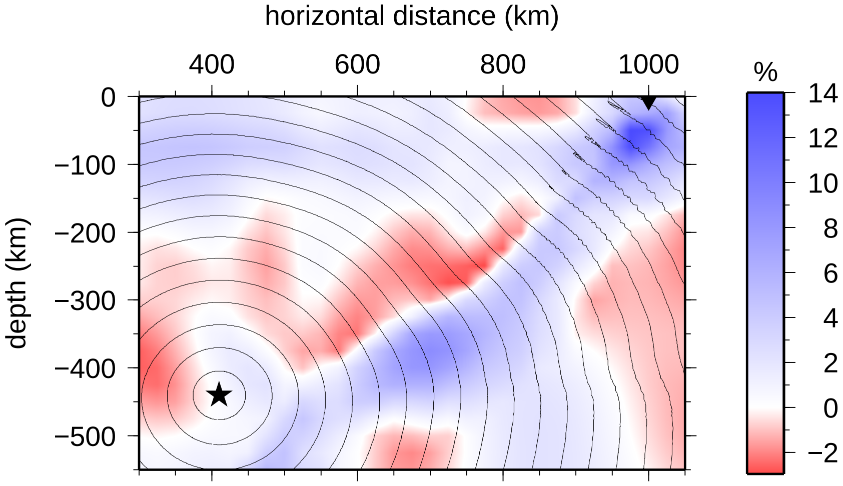


Figure 2.23: Wavefronts computed for a source west of the subducting slab. The grid size for the signed distance function is $226 \times 114 \times 122$.

the spacing in all three dimensions is similar, and 1900 iterations of the update procedure with a time step of 0.005 s are performed. The resulting wavefronts are plotted in figure 2.22. The wavefronts gradually become distorted, and the result overall is not satisfactory despite a computation time of 2 h and 17 min. The wavefronts show similar behaviour as in the wave guide example when the grid resolution was not high enough. The result can only be improved by increasing the grid resolution, but increasing the grid resolution by a factor of two will lead to an increase in computation time by a factor of sixteen.

In the second test, the grid resolution is increased by a factor of two, which leads to a computation time of 36 h and 37 min. The resulting wavefront shows the formation of a swallowtail (see figure 2.23). Considering the computation time, there are serious doubts whether this level set approach is feasible for the routine calculation of multi valued travel times.

The velocity model used here is one of the more complex to which the level set method has been applied so far in seismology, and unlike other studies the simplification of using the paraxial wave equation has not been made. Using the paraxial wave equation means that travel times can only be computed with respect to a preferred propagation direction (e.g. Qian & Leung, 2004, 2006) and this would limit the application of the scheme to reflection seismology or borehole to borehole tomography problems.

The implementation of the level set method chosen here is not based on the latest solvers available (e.g. Losasso et al., 2006; Qian, 2006). By using more advanced concepts for approximating the level set equations (see section 2.2.4), the computational efficiency of this scheme could be improved, especially as the development of numerical methods for the level set equation is a rapidly evolving field. Nevertheless it is doubtful that the reduction in computation time will approach the several orders of magnitude required to make it truly practical.

2.6 Summary

It has been shown in this chapter that an Eulerian approach can be used for the calculation of multi valued travel times. The computation times are, however, too large to consider this technique a feasible method for the prediction of multi valued travel times in practice.

Outside of seismology schemes based on the level set method are often portrayed as an ideal technique for the computation of multi valued travel times (e.g.

Liu et al., 2006; Leung & Qian, 2007). The development of efficient and accurate numerical schemes for the level set equation is a field of ongoing research (e.g. Losasso et al., 2006; Qian, 2006; Frolkovič & Mikula, 2007; Di et al., 2007) and it should be remembered that the method used here does not implement the latest techniques. However, even the most efficient techniques would still need to solve two three dimensional problems in order to find the solution to a two dimensional problem, and they can only resolve features of the two surfaces, and hence of bicharacteristic strip, which extend over several grid cells.

The level set method handles merging and breaking intrinsically. Although not required in seismic wavefront tracking, this superior handling of topological changes, when compared with traditional marker techniques (i.e. Lagrangian), makes the level set method a good choice for fluid dynamic problems (e.g. Sussman et al., 1994; Chang et al., 1996). Nevertheless, one should also keep in mind that the level set method has predominantly been used to evolve a line in two dimensions, and that here the step to three dimensions has been made in order to evolve two surfaces.

The advantage of propagating the bicharacteristic strip instead of the wavefront is that it stays non self-intersecting and locally smooth even for a self-intersecting wavefront with sharp corners. Therefore, when a Lagrangian framework for the propagation of the wavefront is introduced, the phase space representation will be retained. The bicharacteristic strip will be represented in this case by a set of points, and no longer as the intersection of two surfaces given by two three dimensional functions. This has the potential to lead to a far more efficient technique for the computation of multi valued travel times.

Chapter 3

Lagrangian scheme

In the previous chapter the so-called bicharacteristic strip, which represents the wavefront in reduced phase space, is described as the intersection of the zero level set of two signed distance functions. An alternative and arguably more natural approach is to adopt a Lagrangian framework, where the bicharacteristic strip is instead represented by a set of points. As time progresses these points are tracked throughout the medium, with interpolation in reduced phase space applied to retain a uniform density of points. This ensures that the full detail of the wavefront is preserved as it evolves. After presenting the underlying theory, a series of examples are used to demonstrate the capability of the Lagrangian approach to track large numbers of later arrivals in complex two dimensional models.

3.1 Wavefront tracking in reduced phase space

Within a Lagrangian framework, the bicharacteristic strip is represented by a finite set of points. As mentioned previously, for a point source in a two dimensional medium the bicharacteristic strip in reduced three dimensional phase space is given by a line of constant x and y, with $-\pi \leq \theta \leq +\pi$, where (x, y) defines the source location (see figure 3.1). This line in reduced phase space is then represented by a set of points, uniformly distributed in θ. Similarly one can represent an initial plane wave in reduced phase space by a line of constant θ, which is perpendicular to the line used for a point source representation. This line can be discretised by a set of points uniformly distributed in x and y.

The bicharacteristic strip is evolved through the medium in a series of discrete time steps. For a given time step Δt, the strip is updated using a two-stage

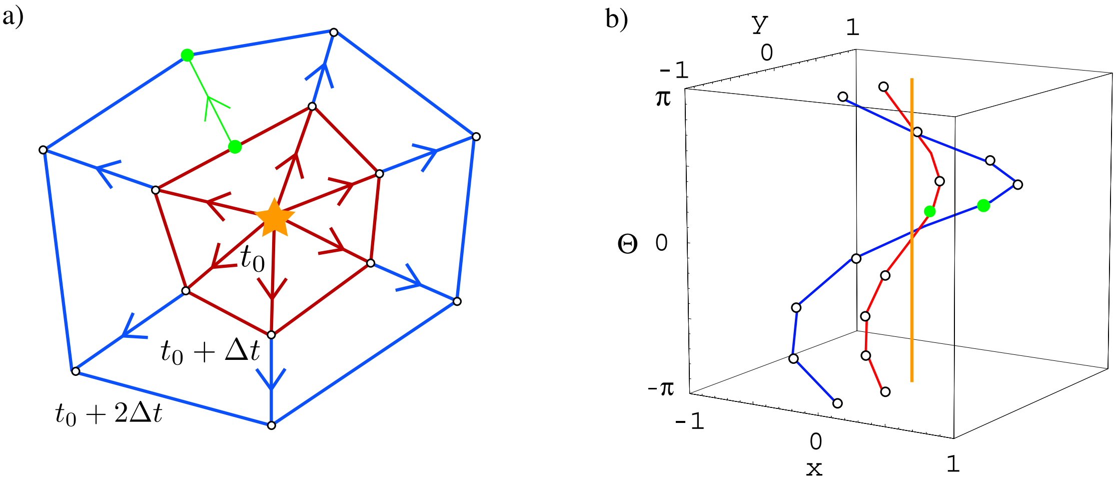


Figure 3.1: Principle of the Lagrangian scheme. (a) Two wavefronts in normal space at different times ($t = t_0 + \Delta t$ in red and $t = t_0 + 2\Delta t$ in blue). (b) Corresponding bicharacteristic strip in phase space. A point (green dot) is inserted if the phase space distance between two neighbouring points is above a certain threshold. This corresponds to the new ray shown in (a). The source location is marked by an orange star in (a) and the initial bicharacteristic strip by an orange line in (b).

procedure. In the first stage, all points are evolved in time by using a fourth order Runge Kutta solver (appendix C) for the following initial value formulation of the kinematic ray tracing equation (e.g. Červený, 2001):

$$\frac{\mathrm{d}x}{\mathrm{d}t} = c\cos\theta,$$
$$\frac{\mathrm{d}y}{\mathrm{d}t} = c\sin\theta,$$
$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = c_x\sin\theta - c_y\cos\theta, \tag{3.1}$$

where $c(x, y)$ defines the wave speed, c_x and c_y are its derivatives in the x and y direction respectively, and θ is the inclination angle of the ray at (x, y). Note that the same system of equations was used in the previous chapter for the grid based solver (see equation (2.43)).

In subsequent examples smooth variations in wave speed $c(x, y)$ are defined by a mosaic of cubic B-spline area elements, the values of which are determined by a regular mesh of control vertices. As mentioned previously, the benefit of cubic B-spline functions is that the first and second derivatives of the resulting field are continuous and given by an analytical expression (see appendix B). Once all points along the bicharacteristic strip at time t have been updated to the new strip at time $t + \Delta t$, points can then be added or removed depending on their separation in reduced phase space (see figure 3.1). The aim is to keep a fixed density of points along the bicharacteristic strip in order to minimise the loss of detail as the wavefront evolves through the medium. As the evolving wavefront distorts in response to velocity heterogeneity, points may also cluster together, resulting in oversampling. It is therefore desirable to allow points to be removed, as it will increase the accuracy to computation time ratio.

If σ represents the initial distance between adjacent points along the bicharacteristic strip, then a new point is added if the distance between two neighbouring points exceeds 2σ, and an existing point is removed if the distance falls below $\sigma/2$. Using a factor of 2 means that the point density will stay close to the initial value during the propagation of the bicharacteristic strip. If, for example, a point were inserted when the separation exceeds 1.1σ, the point density would increase during the wavefront propagation. For the calculation of the distance between two neighbouring points, the two metric coordinates x and y are normalised to lie in the same range as θ (i.e. in $[-\pi, +\pi]$). This scaling allows the norm distance measure in reduced phase space to be defined, and hence the point density.

It is important to keep in mind the periodicity of θ when calculating the distance between two points in reduced phase space. The angle between two wavefront normals can be computed in a clockwise or counter clockwise sense. The angular distance is given by the smaller of the two angles if one assumes that the angular distance between two points along the bicharacteristic strip (i.e. their separation in the θ direction) does not exceed π. Hence, the periodicity of θ poses no problems, as long as the density of points along the bicharacteristic strip is high enough (i.e. more than two points have to be used to represent the bicharacteristic strip for a point source). The periodicity of θ also means that the start and end point of the bicharacteristic strip have to be connected for a point source and one needs to verify if a point is to be inserted or removed between them. In contrast, for a plane wave, the start and end points are not connected.

3.1.1 Local wavefront refinement

Points can be added to the bicharacteristic strip using linear interpolation. If the points along the bicharacteristic strip are numbered using $1 \leq i \leq n$, a point (x^*, y^*, θ^*) can then be inserted half way between the points (x_i, y_i, z_i) and

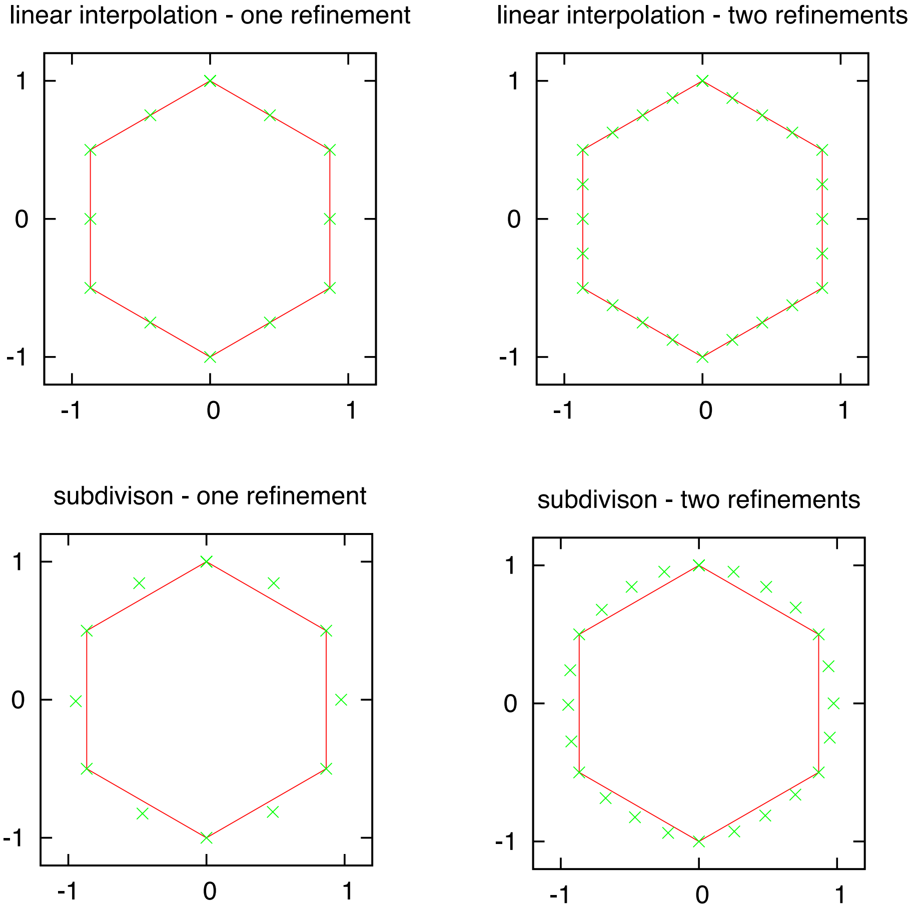


Figure 3.2: Linear interpolation and the higher order interpolation scheme. The initial curve is a hexagon given by the red line. The higher order interpolation will conform to a circle, while the linear interpolation will preserve the hexagon. If one thinks of the hexagon as a seismic wavefront emanating from a point source in a medium with constant velocity, the behaviour of the higher order interpolation is clearly desirable.

$(x_{i+1}, y_{i+1}, \theta_{i+1})$ using the following equations

$$\begin{aligned} x^* &= \frac{1}{2}(x_i + x_{i+1}), \\ y^* &= \frac{1}{2}(y_i + y_{i+1}), \\ \theta^* &= \frac{1}{2}(\theta_i + \theta_{i+1}). \end{aligned} \tag{3.2}$$

A higher order interpolation can be achieved by using the concept of subdivision. The basic idea behind subdivision is to define a smooth curve in two dimensions (or a surface in three dimensions) as the limit of a sequence of successive refinements. This is illustrated for a hexagon in figure 3.2. The shape and smoothness of the resulting curve depends on the rules chosen. Here, an average of nearby points is used: two to the left and two to the right with weights defined as $\frac{1}{16}(-1, 9, 9, -1)$ (Dyn et al., 1990a; Schmalzl & Loddoch, 2003). When applied to a hexagon (cf. figure 3.2) this scheme will eventually delineate a circle after repeated application (Dyn et al., 1990b).

In the following, the term higher order interpolation will be used to refer to a scheme with these weights. It can be shown that the resulting curve has a continuous first derivative if this process is repeated infinitely. The local basis used by the scheme is advantageous, because far away points do not have to be considered when constructing new points. In this case the equations for inserting a point (x^*, y^*, θ^*) half way between the points (x_i, y_i, z_i) and $(x_{i+1}, y_{i+1}, \theta_{i+1})$ are:

$$\begin{aligned} x^* &= \frac{1}{16}(-x_{i-1} + 9x_i + 9x_{i+1} - x_{i+2}), \\ y^* &= \frac{1}{16}(-y_{i-1} + 9y_i + 9y_{i+1} - y_{i+2}), \\ \theta^* &= \frac{1}{16}(-\theta_{i-1} + 9\theta_i + 9\theta_{i+1} - \theta_{i+2}). \end{aligned} \tag{3.3}$$

The results for a model with a constant wave speed gradient will show that the accuracy achieved by using linear interpolation is already sufficient for practical applications. Increasing the initial number of points on the bicharacteristic strip by a factor of two increases the computation time and memory requirements by a factor of only two.

A dynamic data structure is needed to describe the set of points representing the bicharacteristic strip. In this work a linked list (e.g. Metcalf et al., 2004) is used, where each element knows its neighbours and whether it is at the head or tail of the list. If the set of points representing the bicharacteristic strip is stored as a linked list, it is relatively easy to add and remove points. Each point on the bicharacteristic strip is assigned a unique designator. For each time step the points on the wavefront and their connectivity are stored. The designators allow quick access to certain points on the wavefront for a given time step. This becomes necessary when a ray path has to be constructed for a particular arrival.

The idea of explicitly tracking a wavefront by advancing a set of points using local ray tracing and interpolation has been investigated over the last decade and half, mainly in the exploration seismology field (e.g. Vinje et al., 1993; Lambaré et al., 1996; Buske & Kästner, 2004). However, in these earlier studies, ray density has been defined only in normal space - for example, the metric distance between

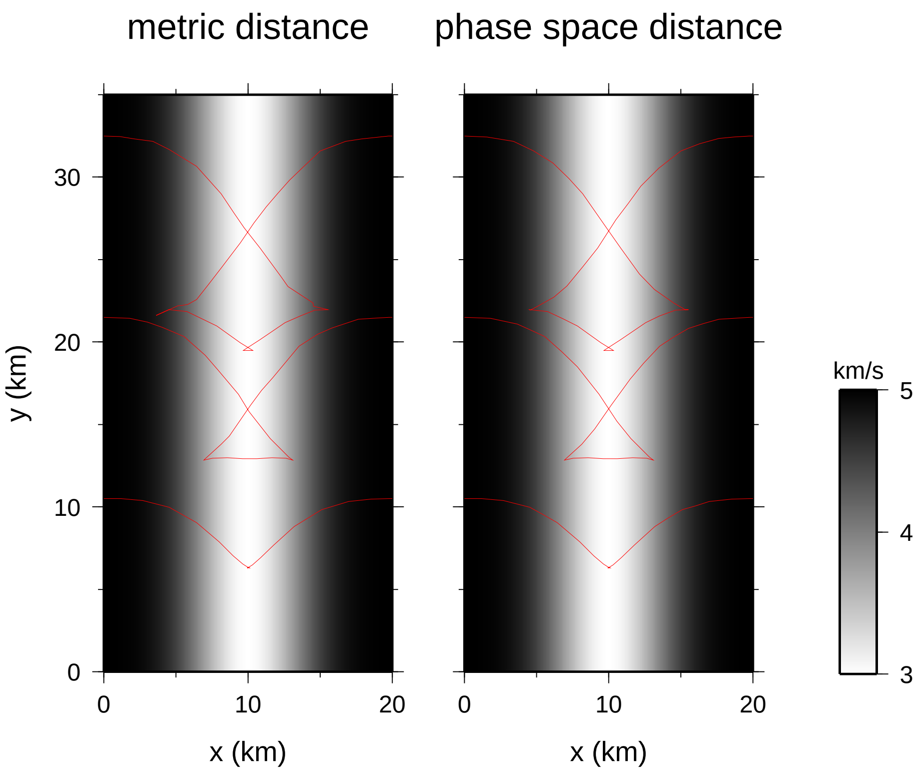


Figure 3.3: A plane wave parallel to the x-axis at $y = 0$ propagates in the positive y-direction. The wavefront begins to triplicate and forms a swallowtail due to the low velocity in the centre. When the reduced phase space distance is used as a criterion for adding points (right), the swallowtail pattern is better recovered, compared to when the metric distance is used (left). The initial plane wave is in both cases represented using 25 points.

neighbouring rays (Lambaré et al., 1992; Vinje et al., 1993; Ettrich & Gajewski, 1996) or the angular distance (Sun, 1992). These definitions of ray density tend to encounter difficulties if the wavefront starts to develop a swallowtail pattern; they are only loosely correlated to the complexity of the ray field. The use of a phase space distance metric is a key element of the Lagrangian method employed here and is superior to the alternative of using a metric defined in normal space. Lambaré et al. (1996) use a similar criterion for the ray density in phase space within their Hamiltonian formulation of ray theory. Figure 3.3 illustrates the advantage of using the concept of a phase space distance for the refinement. Here a plane wave enters the medium from the bottom and propagates to the top. Clearly the end points of the swallowtail are much better recovered when the distance in reduced phase space is used to decide when to add or remove a point, compared to when the metric distance is used.

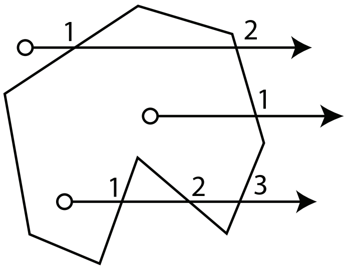


Figure 3.4: Illustration of the crossing method. One begins at the point in question and moves along the x axis in the positive x direction counting how many edges are crossed. If the number of edges crossed is odd the point lies outside the polygon and if it is even the point lies inside the polygon.

3.1.2 Extracting arrival information

In most practical applications, the source-receiver travel time, and in many cases the associated ray paths, are needed rather than the wavefront locations. The main challenge in extracting this information is to locate the receiver in an irregular and potentially multi valued travel time field. Since the wavefront is explicitly defined at each time step, it is possible to identify the two consecutive wavefronts at times t and $t + \Delta t$ and the adjacent ray paths that together bound a receiver. Using the two wavefronts and the associated ray path segments between the points on the wavefront a set of adjacent polygons can be defined. The problem of calculating an arrival time at a receiver then becomes one of identifying the polygon in which a receiver is located. Ettrich & Gajewski (1996) suggest an approach based on using the products of vectors pointing from the receiver to the four corners of the polygon in order to determine whether or not a receiver is located within the polygon. If all of the resulting vector products have the same sign the receiver is considered to be in the polygon.

On the other hand, testing whether a point lies inside a polygon is a basic operation in computer graphics, and a wide variety of efficient algorithms have been discussed by Haines (1994). The approach adopted in this thesis is the so called crossing method (Haines, 1994), where one traces a line from the receiver horizontally (increasing x and constant y) and counts how many edges it crosses (see figure 3.4). If the number of edges crossed is even, the point lies outside the polygon; if it is odd, the point lies inside the polygon. If it has been determined that a receiver lies inside the polygon, an arrival time needs to be calculated. The time and position of the two wavefront segments of the polygon is known and so the shortest distance from the receiver to each wavefront can be calculated. Using these two distances, an arrival time can be linearly interpolated. This scheme is similar to an approach advocated by Vinje et al. (1993). The search for a receiver needs to be completed before points are added or removed from the wavefront at time $t + \Delta t$, otherwise the set of points on both wavefronts may not correspond.

In addition to travel time, it can also be useful to locate the ray path for each arrival, especially for applications such as seismic tomography. A source-receiver ray path can be constructed in normal space by following the wavefronts back from the receiver to the source, once the wavefront propagation is finished (see figure 3.5). This requires that for all wavefronts, the points and their connectivity have been stored during the propagation process. For a given arrival at a receiver, the two neighbouring ray segments are known and the closest point to the receiver on the wavefront segment at time t is also known. The relative distances $d_1/(d_1 + d_2)$ and $d_2/(d_1 + d_2)$ to the two neighbouring points on the wavefront are then calculated. In the next step, the wavefront at time $t-\Delta t$ is used. The neighbouring points based on the wavefront at time t and the relative distances are known, so this information can be used to calculate the position of the ray path at time $t - \Delta t$. It is then possible to step back in time until one reaches the source.

While the wavefront tracking is performed in reduced phase space, the back tracking procedure is performed in real space. The θ values of the nodes on the wavefront are only used when the takeoff angle for the ray is interpolated. Reduced phase space is used during wavefront tracking in order to have a superior criteria for maintaining a fixed density of points on the wavefront. Once the wavefront tracking is finished, a set of wavefronts (i.e. isochrons of the travel time field) is given. The assumption is that the point density on these wavefronts is high enough to back track in real space using a straight line for the ray segment between the wavefronts. If more accurate ray paths are needed one could compute a path between the two consecutive wavefronts using an interpolation scheme which also takes the direction of the local wavefront normal θ into account. However, we note that the accuracy of the ray paths extracted in real space has not turned out to be an issue for any of the applications presented in this thesis.

When stepping back in time from the current wavefront to a previous wavefront, points may appear or disappear from the wavefront. This is where the connectivity of the points can be used to find a new set of neighbouring points for the current ray path. If a point no longer exists, one can simply check if its neighbour exists, and then use this point after having recalculated the relative distances. It is also possible that a new point closer to the ray path appears. Again,

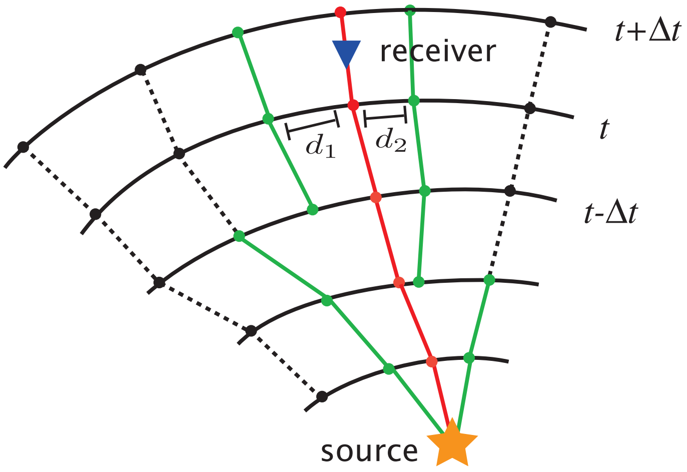


Figure 3.5: Determination of a ray path using the wavefronts calculated by the Lagrangian solver. A ray path (red line) for a receiver is interpolated back to the source between the known neighbouring ray path segments (green lines).

connectivity information is used to identify these points, and the relative distances are recalculated.

During the back tracking procedure, only a few specific nodes on each wavefront are needed. This requires careful book keeping of the wavefronts during the propagation phase, and an efficient scheme for accessing the required nodes of each wavefront during the back tracking phase. Preferably, wavefront information is stored in memory, but for large problems (e.g. in the next chapter), it is stored on disk. Therefore, the resulting algorithm tends to be more complex when compared with a ray extraction scheme that follows the gradient of the first arrival travel time field (e.g. Rawlinson & Sambridge, 2004a). However it allows the extraction of ray paths for all arrivals rather than just for the first arrivals.

3.1.3 Dynamic ray tracing

The arrival time alone is probably not sufficient to identify later arrivals in most cases. Additional information in the form of amplitude estimates from geometrical spreading will help to discriminate between the various incoming phases. Paraxial ray theory can be used to obtain the coefficients required for the calculation of geometrical spreading (e.g. Vinje et al., 1993; Červený, 2001). While the ray path is described by the kinematic ray tracing equations, information about the wavefront in the vicinity of the ray like the geometrical spreading and the wavefront curvature can be obtained through the dynamic ray tracing equations. They are given by (e.g. Vinje et al., 1993; Červený, 2001)

$$\frac{\mathrm{d}q}{\mathrm{d}t} = v^2 p, \tag{3.4}$$

$$\frac{\mathrm{d}p}{\mathrm{d}t} = -\frac{v_{\mathbf{nn}}}{v} q, \tag{3.5}$$

where q denotes the geometrical spreading factor, also known as the Jacobian in ray theory literature (e.g. Červený et al., 1977; Červený, 2001), p the auxiliary function, v the velocity and $v_{\mathbf{nn}}$ the second derivative of v in the direction of the ray perpendicular (i.e. tangent to the wavefront) vector $\mathbf{n}$. Given a travel time field T, the vector $\mathbf{n}$ can be defined using the dot product $\mathbf{n} \cdot \nabla T = 0$. The second derivative of the velocity field in the direction of $\mathbf{n}$ can be written as.

$$\begin{aligned} v_{\mathbf{nn}} &= \nabla\left(\nabla v \cdot \mathbf{n}\right) \cdot \mathbf{n} \\ &= n_x^2 \frac{\partial^2 v}{\partial x^2} + 2 n_x n_y \frac{\partial^2 v}{\partial x \partial y} + n_y^2 \frac{\partial^2 v}{\partial y^2}, \end{aligned} \tag{3.6}$$

where n_x and n_y are the x and y components of the ray perpendicular vector $\mathbf{n}$. The ray perpendicular vector can be easily constructed if the direction of the local ray is known. The geometrical spreading factor q for a given ray can be calculated using a fourth order Runge Kutta scheme (appendix C) to solve (3.5) numerically along the ray. Once this has been obtained an amplitude coefficient can then be computed using

$$A(t + \Delta t) = A(t) \sqrt{\frac{\rho(t + \Delta t) v(t + \Delta t) q(t)}{\rho(t) v(t) q(t + \Delta t)}}, \tag{3.7}$$

where ρ denotes the density. In the following, only two dimensional problems are considered but out of plane spreading is taken into account. The assumption used for the out of plane spreading is that the velocity does not change in the direction perpendicular to the profile. The initial conditions for a line source perpendicular to the plane are given by (e.g. Červený, 2001)

$$q = 0 \quad \text{and} \quad p = \frac{1}{v}. \tag{3.8}$$

To obtain the the geometrical spreading factor q for a point source, the factor for the line source is multiplied by a correction function $q_\perp$ (e.g. Červený & Hron,

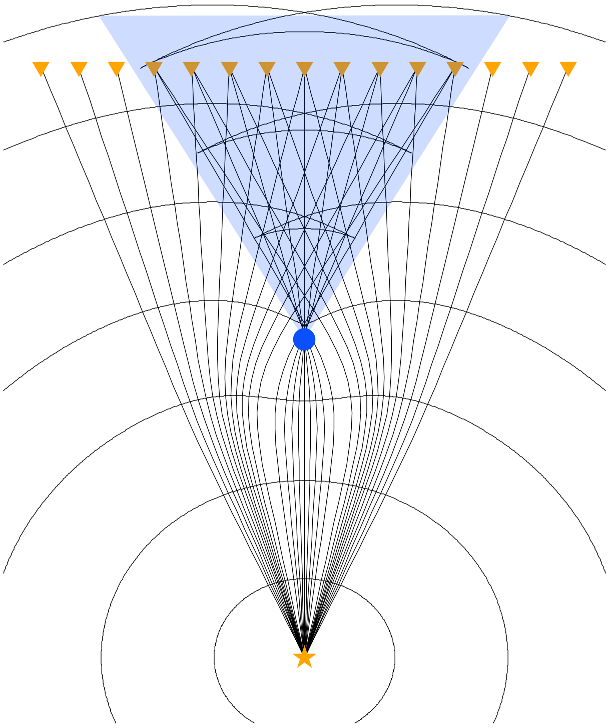


Figure 3.6: Ray paths and wavefronts in the presence of a low velocity anomaly. The source is given by the the orange star and the receivers are marked by the orange triangles. The caustic point is marked by the filled blue circle and the blue shaded zone marks the region where the wavefront triplicates.

1980; Červený, 2001), which describes the out of plane spreading:

$$q_{\perp} = \frac{1}{v} \int_0^S v(s)\, \mathrm{d}s, \tag{3.9}$$

where s is the path length along the ray from the source to the receiver and the source-receiver distance along the ray path is given by S.

One way to calculate the amplitude coefficient involves solving the dynamic ray tracing equation during the wavefront propagation phase. However, in practice, amplitude information is only required at a limited set of receivers and not for the whole wavefront. It is therefore more efficient to calculate the amplitude coefficient only for paths associated with each receiver. Once the ray path for an arrival is known, the dynamic ray tracing equations can be solved along this path using a fourth order Runge Kutta scheme (appendix C). The absolute value of the amplitude coefficient is not needed; rather it is the relative amplitude of the different arrivals which is used to aid identification in real data.

In complex media the geometrical spreading factor q can become zero and change its sign when the ray paths begin to intersect one another. These points or lines are known in the literature as caustics (e.g. Červený, 2001). Figure 3.6 illustrates the formation of caustics in two dimensions for a simple low velocity anomaly.

This behaviour of q is understandable when one thinks of geometrical spreading as the change in distance between two neighbouring ray paths. As the two ray paths get focused in a low velocity zone, they become closer together and finally intersect each other. At their intersection point the distance between them is zero. If the distance is defined with respect to the ray perpendicular vector $\mathbf{n}$, the sign of the distance is changed as the rays begin to diverge. However, this effect can lead to a singularity in the amplitude (3.7), as it is no longer possible to compute an amplitude coefficient for a ray if q is zero for the time $t + \Delta t$. Changing the size of the time step so that $q(t + \Delta t)$ differs from zero can help to avoid this problem during the computational process provided the caustic point is not close to the receiver. In chapter 4 synthetic seismograms will be computed using the Gaussian beam method which can deal with caustics and therefore phase distortions automatically.

3.2 Examples

Wavefront construction schemes which use similar principles to the one presented here have previously been introduced (e.g. Vinje et al., 1993; Lambaré et al., 1996; Lucio et al., 1996). However, in addition to the phase space approach and several other refinements adopted here, a further distinguishing feature of this research is that the Lagrangian solver is also tested in solid earth applications and not just exploration, which has been the traditional focus.

The first example involves a constant velocity model. This allows a comparison between the results of the Lagrangian scheme and those of the Eulerian scheme presented in the previous chapter (figure 2.18). A constant velocity gradient model is then introduced to investigate the effects of using different point densities and interpolation schemes to describe the evolving wavefront. Next we demonstrate how the scheme can successfully calculate later arrivals for a subduction zone model and the Marmousi model. In the last example, the method is used to track complex multi valued surface waves. The scheme has been implemented under GNU/Linux in Fortran and all computation times are given for a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory.

number of starting points	number of time steps	size of the time step (s)	computation time (s)
50	100	0.1	4.92
100	200	0.05	16.29

Table 3.1: Computation time for the two configurations used in the first test.

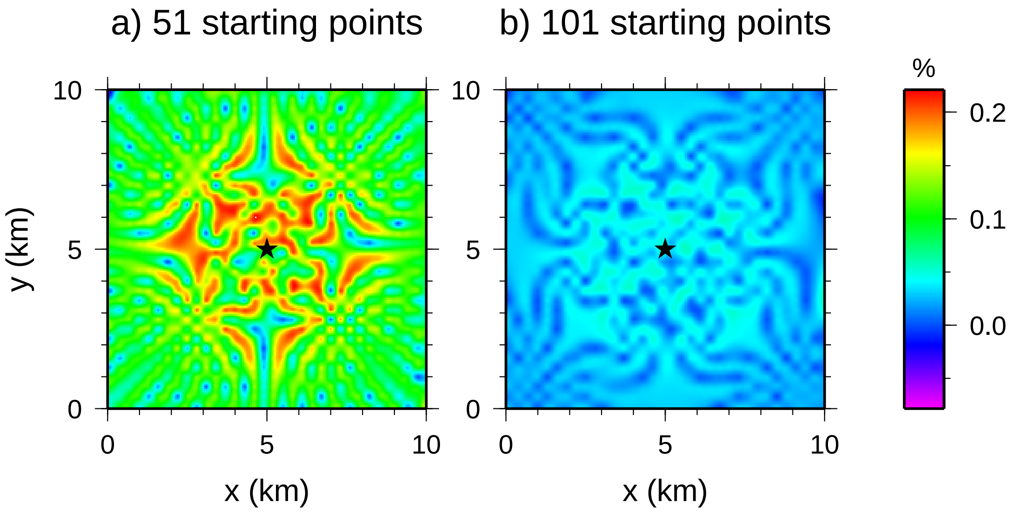


Figure 3.7: Relative error in percent between the analytical and numerical solution for the two configurations given in table 3.1. The source location is marked by the black star. The overall RMS error in (a) is 0.122 % and in (b) 0.0306 %

3.2.1 Constant velocity

A point source is placed in a medium with a constant velocity of 1 km/s. This is a simple model, but it can be used to map out the error pattern and estimate the maximum achievable accuracy. The travel times are calculated using different point densities and time steps (see table 3.1).

In figure 3.7 the relative error between the analytical and numerical solution is given in percent for the the two configurations. The source is located in the centre. As one would expect, increasing the initial number of points on the bicharacteristic strip decreases the overall error. For both configurations, the error is significantly smaller than the chosen time step. The error pattern shows some symmetry and exhibits a significant variability. The RMS error is 0.122 % if the initial number of points is 50 and 0.0306 % if the initial number of points is 100. Doubling the initial number of points and halving the time step leads, as expected, to an increase in accuracy by a factor of four.

A potential source of error is the assumption that the wavefront segments between the points on the wavefront are straight lines. This assumptions limits the accuracy of the points being inserted during the propagation phase. It also limits the accuracy of the extracted travel times. Using the linear interpolation scheme to estimate the arrival time at a receiver within a polygon is another potential source of error.

The RMS errors for the two configurations is slightly larger than in section 2.5.1. However, one should keep in mind that the simplest possible Lagrangian scheme is used here: The linear interpolation of new nodes to the wavefront means that no unfair advantage is given to the Lagrangian scheme. On the other hand, computation time is only a fraction of what is required for the Eulerian approach. In other words, we can compute a set of travel times eight times with an RMS error of 0.0306 % using a Lagrangian scheme in the time it takes to compute a set of travel times once with an RMS error of 0.087 % using an Eulerian solver. Clearly, the Lagrangian scheme is going to be a much more efficient tool for the calculation of multi valued travel times.

3.2.2 Constant velocity gradient

This example is used to investigate the accuracy of the Lagrangian scheme with respect to the size of the time step, the number of points on the bicharacteristic strip and the interpolation scheme. In a medium with a constant velocity gradient, the ray trajectories are given by circular arcs (see figure 3.8) and the analytical solution for the travel time between a source and a receiver at the surface is given by (Sheriff & Geldart, 1995)

$$i_0 = \tan^{-1}\left(\frac{2v_0}{ax}\right), \tag{3.10}$$

$$t = \frac{2}{a}\ln\left(\cot\left(\frac{i_0}{2}\right)\right), \tag{3.11}$$

where a is the velocity gradient, x the horizontal distance to the receiver, v_0 the velocity at the surface and t the travel time. In this test 39 receivers are placed at the surface and the velocity gradient is $0.15\,\mathrm{s}^{-1}$.

A large set of travel times to the receiver array were calculated by varying the time step, initial numbers of points on the bicharacteristic strip and interpolation scheme. The mean and maximum relative error in percent for the 39 receivers as well as the computation time are given in figures 3.9 and 3.10. In figures 3.11

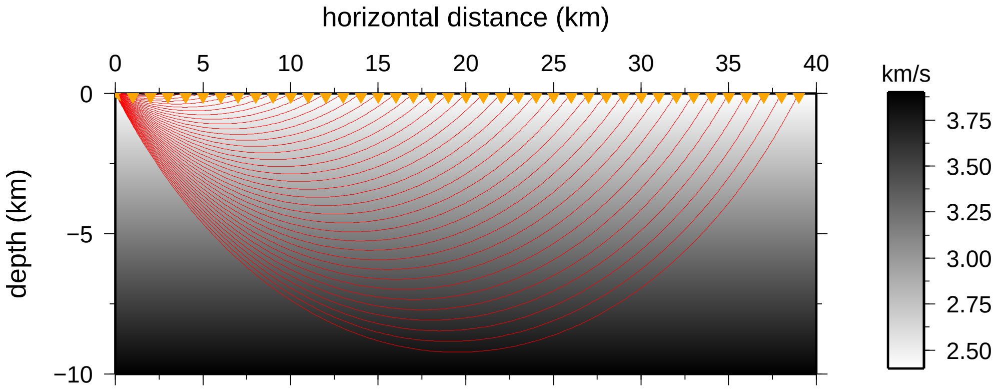


Figure 3.8: Ray paths for a model with a constant velocity gradient of $0.15\,\mathrm{s}^{-1}$. The source is located in the upper left corner and the orange triangles denote receivers.

and 3.12 the relative travel time error is plotted along the line of receivers for four selected combinations of time step and initial number of points.

There are two ways to increase the accuracy: one can reduce the size of the time step or increase the number of points on the bicharacteristic strip. Increasing the number of initial points on the bicharacteristic strip has overall a stronger influence than reducing the time step (see figure 3.9 and 3.10). For time steps with a size above 0.5 s one observes a significant decrease in accuracy for a constant number of points. As far as the computation time is concerned, increasing the initial number of points has almost no impact when compared to reducing the time step. A high density of points on the bicharacteristic strip reduces the error in two ways. First, when points are added to the wavefront some form of interpolation has to be used. If the point density is high, the error due to the interpolation tends to be smaller. Second, when computing a travel time to a receiver, a higher density of points means that the distance over which the travel time has to be interpolated is smaller. This seems to be less of a factor when the time step is varied. In figure 3.9 linear interpolation is used for adding new points while in figure 3.10 the previously introduced higher order interpolation scheme is used. In the latter case the error is smaller, without a significant increase in computation time. Nevertheless, these results show that the accuracy achieved with linear interpolation is of the same order of magnitude as that associated with the higher order scheme.

Figure 3.11 and 3.12 illustrate that the error is not constant along the line of receivers. An explanation for the oscillation of the error could be that the distance over which the travel time has to be interpolated at a receiver also varies from one receiver to another.

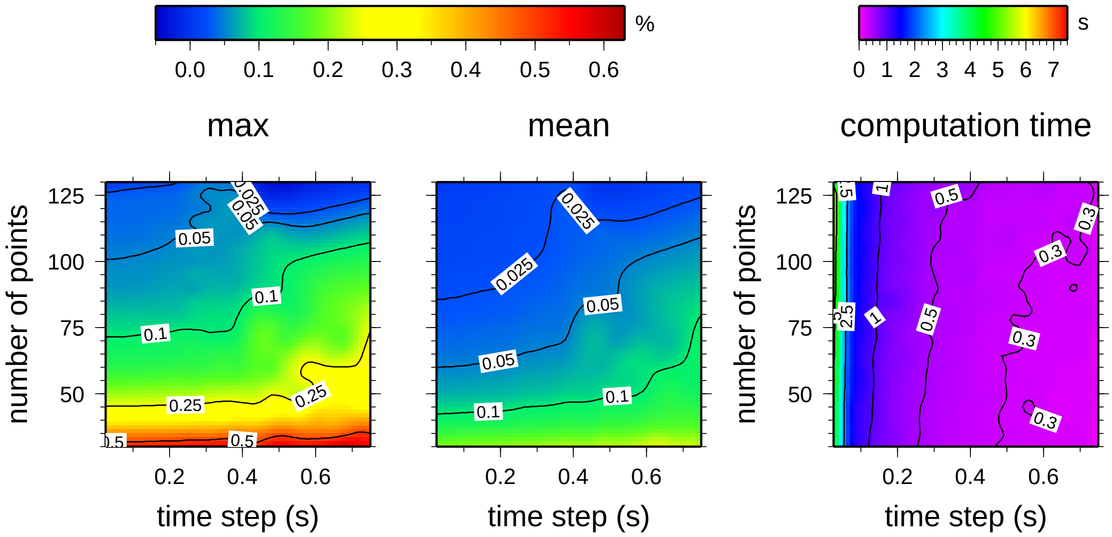


Figure 3.9: Difference between the analytical and numerical solution for different time steps and initial numbers of points on the bicharacteristic strip. Linear interpolation is used in reduced phase space for adding new points.

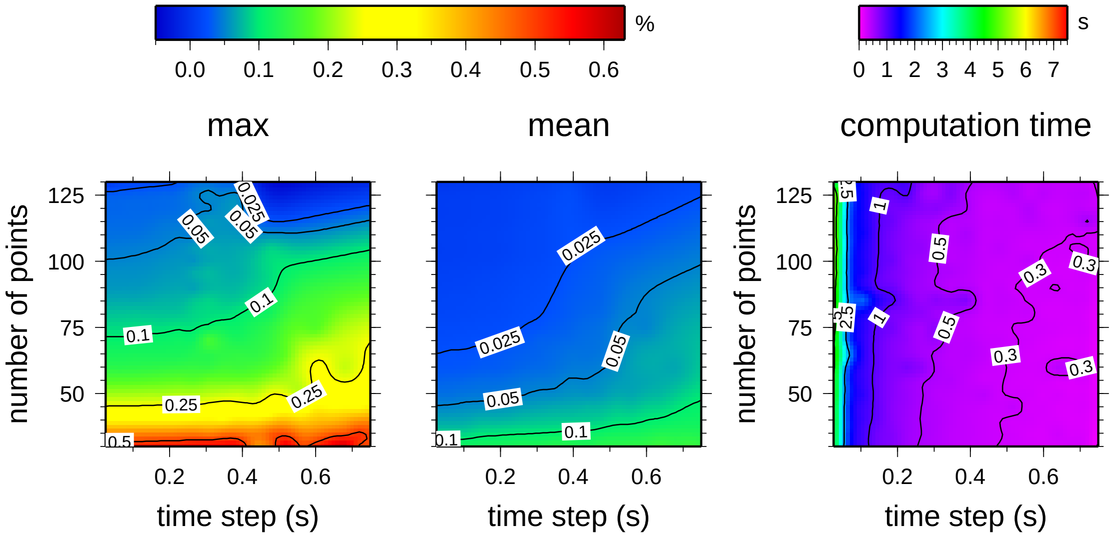


Figure 3.10: Difference between the analytical and numerical solution for different time steps and initial numbers of points on the bicharacteristic strip. The higher order interpolation scheme (section 3.1.1) is used in reduced phase space for adding new points.

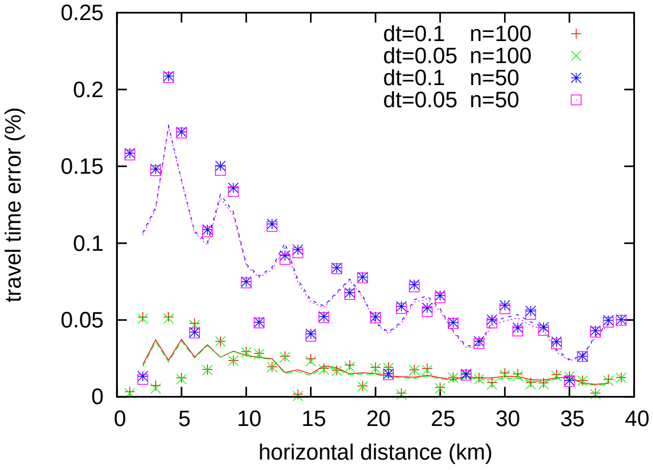


Figure 3.11: Difference between the analytical and numerical solution for different time steps (dt) and initial number of points on the bicharacteristic strip (n). Linear interpolation is used in reduced phase space for adding new points.

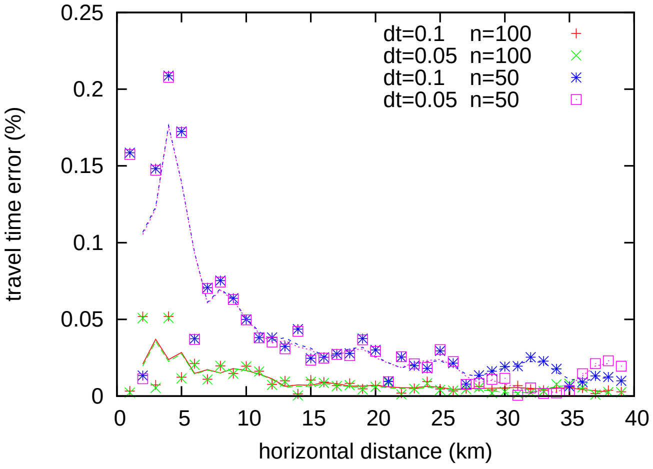


Figure 3.12: Difference between the analytical and numerical solution for different time steps (dt) and initial number of points on the bicharacteristic strip (n). The higher order interpolation scheme (section 3.1.1) is used in reduced phase space for adding new points.

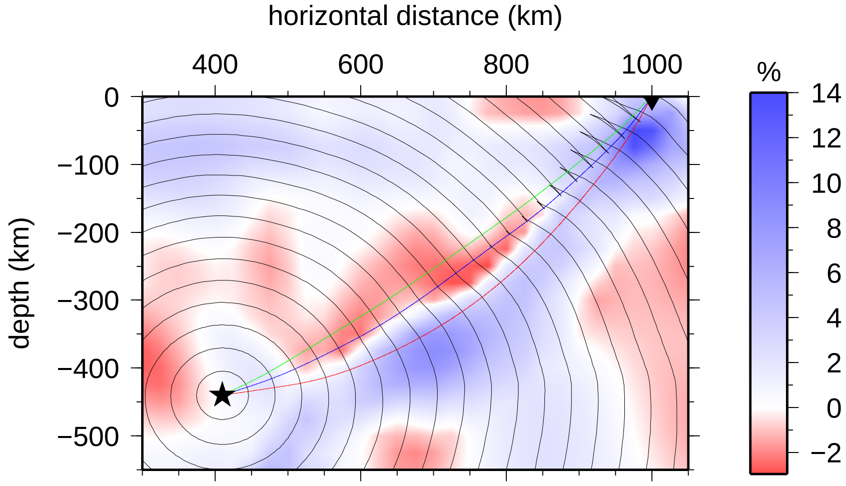


Figure 3.13: Wavefronts computed for a source west of the subducting slab imaged by Conder & Wiens (2006) in the Fiji-Tonga region. Compared with the wavefronts calculated using the Eulerian scheme (see figure 2.23) the triplication is much better recovered. The ray paths corresponding to the first, second and third arrival are plotted as red, green and blue lines, respectively. In this and the following figures involving this model, the speed is plotted as a perturbation with respect to a locally derived one dimensional model (Conder & Wiens, 2006).

3.2.3 Subduction zone

In the previous chapter, when the wavefront was tracked through the subduction zone velocity model for the Tonga Fiji region (e.g. figure 2.23) using the Eulerian scheme, a triplication developed when a higher grid resolution was used. The wavefront calculated for the same source point using the Lagrangian scheme with a time step of 0.05 s and 150 points on the initial bicharacteristic strip is shown in figure 3.13. As before, the model has been up-sampled twice using a bilinear interpolation so that the grid spacing is 12.5 km. Based on this result, the Lagrangian scheme is again clearly preferable to the Eulerian approach. The triplication in this case appears to be recovered with much greater detail (compared with figure 2.23) and the computation time is less than 5 s, which compares to several hours for the Eulerian scheme.

One way to investigate the behaviour of later arrivals with respect to a given model is to search for regions where earthquakes are likely to generate later arrivals at a receiver. In this case the velocity field is up-sampled ten times using a bilinear interpolation in order to avoid over smoothing of the model due to the cubic B-spline approximation used in the Lagrangian scheme. The P-wave velocity model is now given by 561 × 281 nodes spaced 2.5 km apart horizontally and vertically.

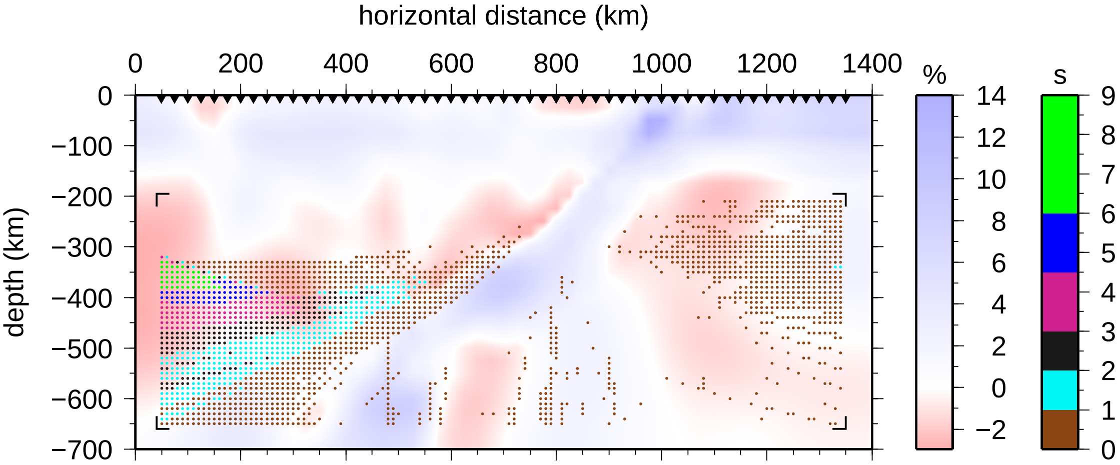


Figure 3.14: The coloured points represent sources capable of generating later arrivals at one of the surface receivers (black triangles). The points are coloured according to the maximum travel time difference between the first and second arrival at any given receiver. The black corners represent the boundary of the region within which the search for sources which generate multi arrivals was performed.

The reason why the model has not been up sampled ten times for the previous example (see figure 3.13) is that the intention was to compare the results with the those obtained using the Eulerian solver (section 2.5.3).

Figure 3.14 shows sources which generate later arrivals at one or more receivers. This figure is computed by evaluating 5850 potential source points in 24 hours, which means that each source point is evaluated in about 15 s. This execution time can be compared to the Eulerian solver, which took longer to compute the travel time field for a single source than the Lagrangian scheme did for all 5850 sources.

It is important to realise that the search for the presence of multi arrivals is performed in a model obtained from seismic tomography using the travel times of first arrivals. This may mean that later arrivals are under-represented, as first arrival tomography tends to under-estimate the amplitude of low velocity anomalies. This is because first arrival ray paths avoid regions with low velocities (see figure 3.13). Due to the convolving effects of the source wavelet and earth response to the underlying wavetrain, as well as the dominant frequency, it may well not be possible to distinguish between most of these arrivals.

Note that only sources located to the left (i.e. west) and above the subducting slab have generated later arrivals with a significant delay, when compared to the first arrival. The scattering in the distribution of sources that generate later

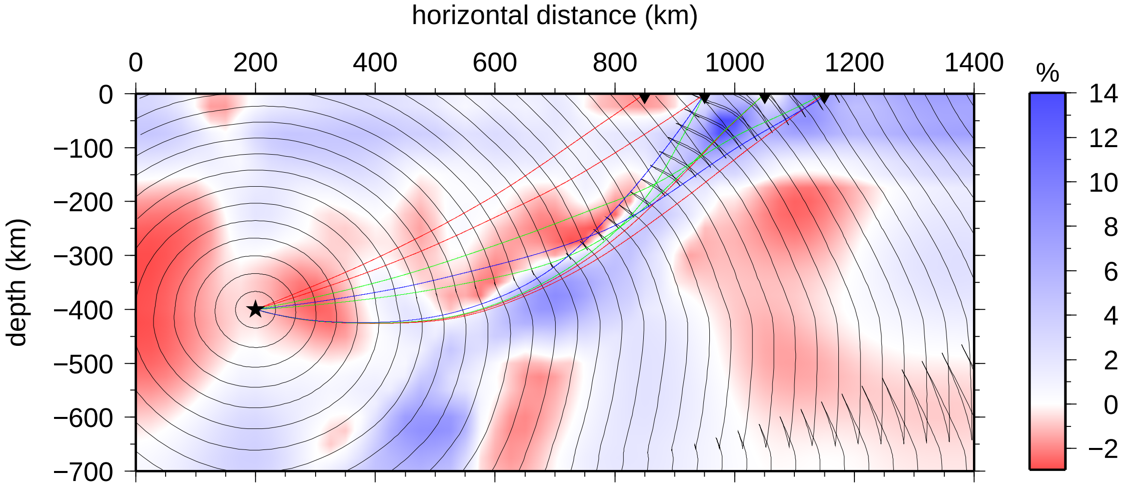


Figure 3.15: Wavefronts computed for a source causing a significant difference (5.0 s) in the arrival times between the first and second arrival for the receiver at 950 km horizontal distance. The ray paths corresponding to the first, second and third arrivals are plotted as red, green and blue lines respectively .

arrivals with a travel time delay less than 1.0 s is partly due to a finite number of receivers used. It is possible that a small swallowtail pattern, corresponding to a small travel time difference between the first and later arrival, might arrive at the surface between two receivers. Therefore no receiver registers this later arrival.

The closer the structure initiating a swallowtail pattern is to the receiver, the smaller the difference in arrival time due to multipathing (see figure 3.13). A low velocity structure close to the source and strong enough to bend the wavefront inwards is sufficient to generate large differences in the arrival times at many receivers far away from the source (see figure 3.15). Generally, the generation of multiple arrivals is very sensitive to small changes in the velocity structure, especially if they are close to the source point. Only a few sources are capable of generating later arrivals in regions where earthquakes are actually observed, which is close to the subducting slab. In figure 3.14 the sources which generate large travel time differences between first and second arrivals are located in regions where no earthquakes are observed. Figure 3.15 shows an example of a source-receiver combination where there is a significant difference in arrival time between the first and second arrival at the receiver at 950 km horizontal distance. While no earthquakes have been observed close to this particular source, it nevertheless shows how the inclusion of later arrivals could in principle improve coverage, since ray paths for the two later arrivals sample different parts of the model compared to the first arrival path.

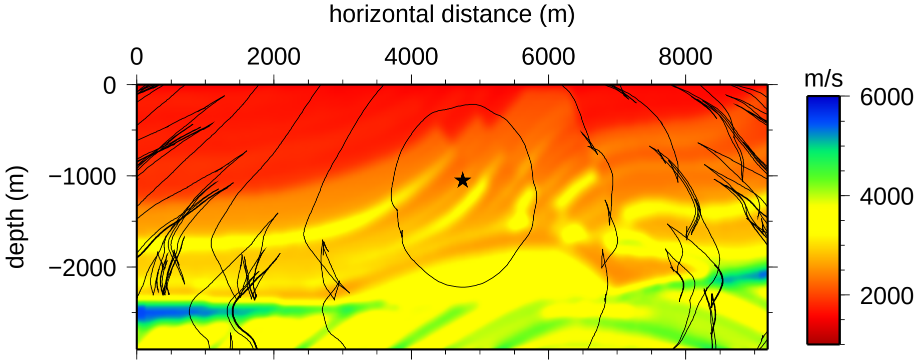


Figure 3.16: Smooth version of the Marmousi model. To obtain the smooth version the original model has been convolved with a spatial Hanning (cos^2) filter of radius 150 m. A source is located at (4750,-1050) and the associated wavefronts are plotted at 0.4 s intervals.

3.2.4 Marmousi model

The Marmousi model (figure 3.16) is often used to demonstrate the limitations of first arrival travel times in the imaging of complex media. Geoltrain & Brac (1993) showed that multi arrival travel times are needed in order to accurately image the Marmousi model. The underlying geological structure is based on a profile through the North Quenguela trough in the Cuanza basin in Angola (Versteeg, 1993). Traditionally, travel times have been computed using a smoothed version of the Marmousi model (e.g. Buske & Kästner, 2004; Coman & Gajewski, 2005; Qian & Leung, 2006). The smooth model was obtained by convolving the so-called hard model, which is characterised by strong velocity gradients, with a spatial Hanning (cos^2) filter of radius 150 m (see `http://www.rocq.inria.fr/~benamou/testproblem.html`). The model consists of 384 × 122 nodes with a vertical separation of 24 m.

Figure 3.17a illustrates ray paths and relative amplitudes obtained by the Lagrangian solver for the standard test problem, using the smoothed version of the model with a source at $(6000, -2800)$. The scheme is able to predict 651 arrivals and the corresponding ray paths for the 384 evenly spaced receivers positioned at the surface. The computation time in this case is 45 s using 150 points on the initial bicharacteristic strip and a time step of 0.002 s. More than 40 % of the arrivals are later arrivals, which tend to sample regions avoided by first arrivals. Figure 3.17b shows the ray paths for a source close to the left boundary of the model, at $(50, -2600)$. As the wavefronts for this source position travel a greater distance, they become much more complex and the fast region acts as a kind of

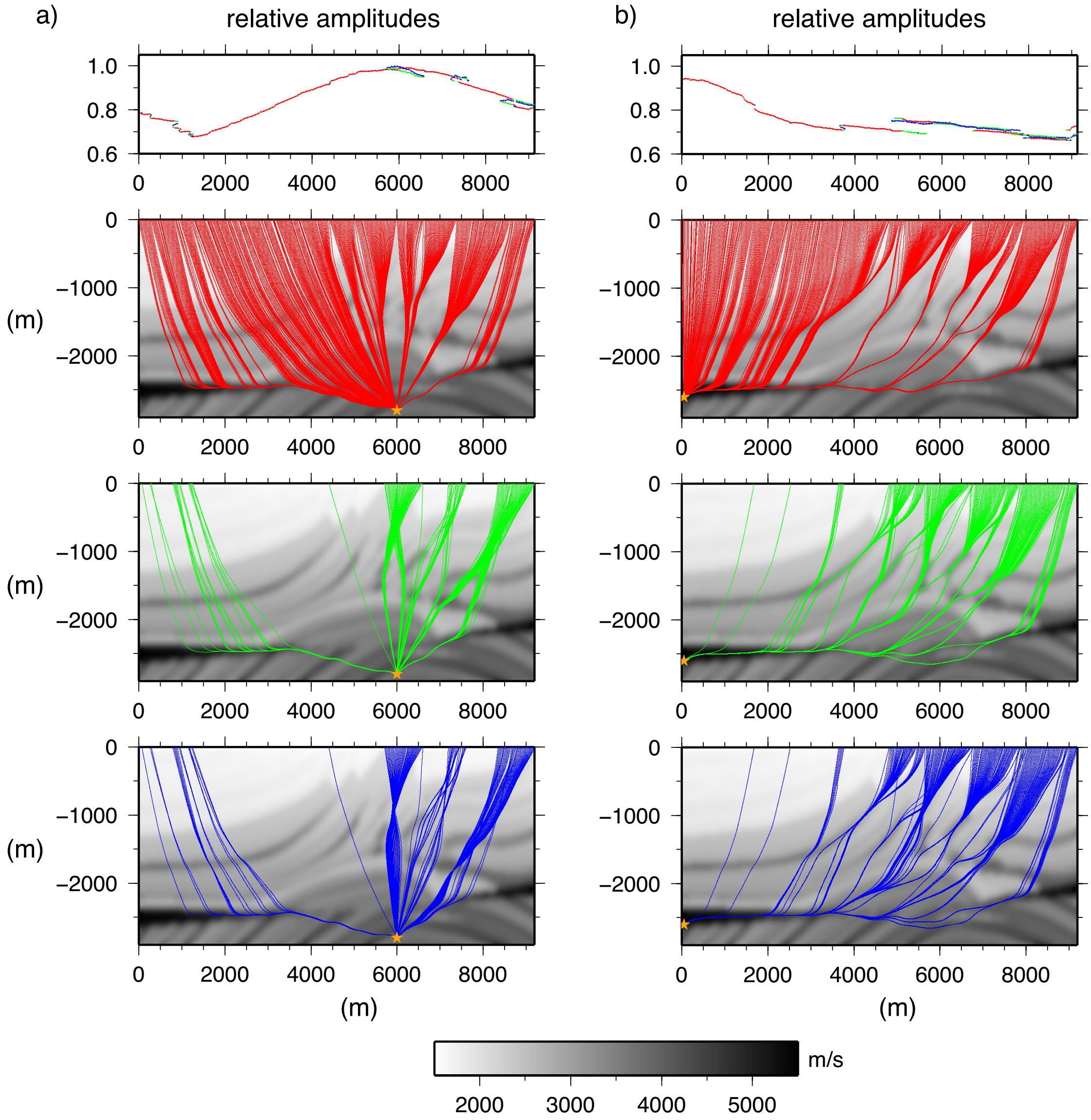


Figure 3.17: Multiple arrivals and amplitude information for two sources, (a) and (b), in the Marmousi model. The ray paths for the first arrivals are given in red, for the second arrivals in green, and for third and later arrivals in blue. The figures in the top row show the relative amplitudes of the first (red), second (green), and third arrival (blue).

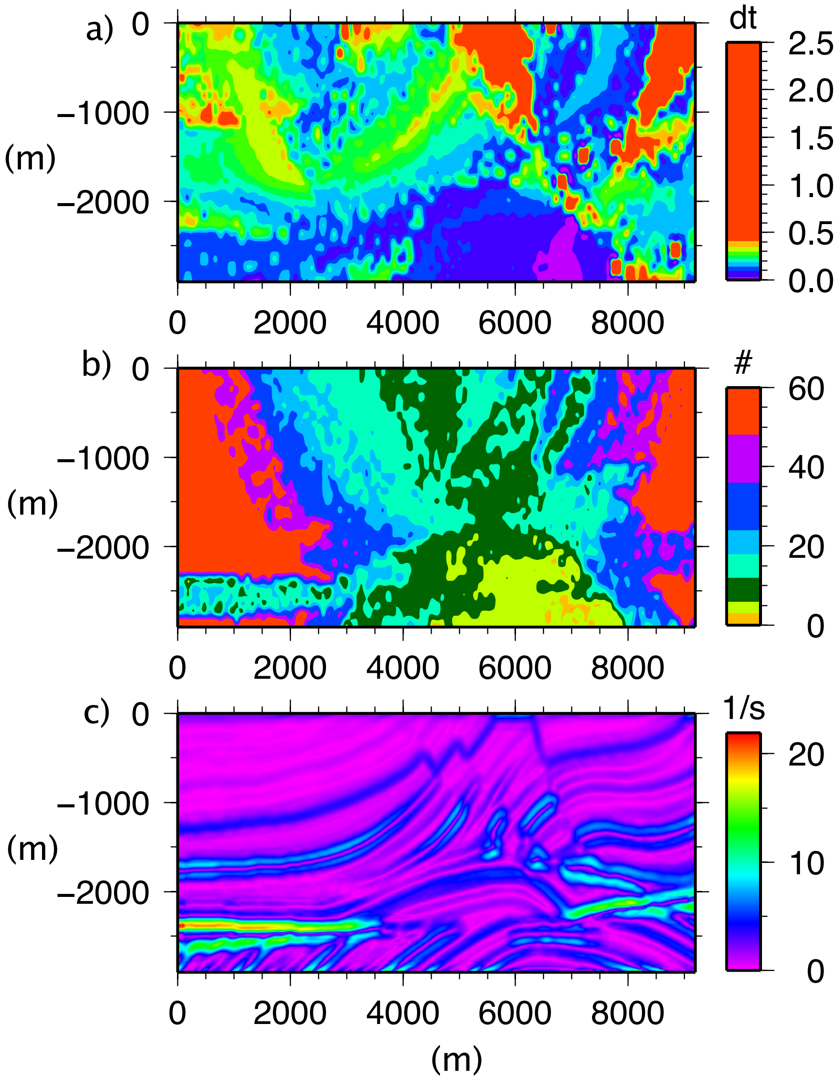


Figure 3.18: (a) Maximum observed travel time difference between first and second arrival for any one of the receivers at the surface as a function of source location. (b) Maximum number of arrivals for any one of the receivers at the surface as a function of source location. (c) Velocity gradient of the Marmousi model.

wave guide. The scheme predicts 3291 arrivals for this example, with later arrivals accounting for more than 85 % of the total. For both source locations, the later arrivals tend to sample regions avoided by the first arrivals. Dynamic ray tracing is used to calculate relative amplitudes of the first, second, and third arrivals. The later arrivals have similar, and for some receivers, larger amplitudes than the first arrival. This confirms that the first arrival of a wavefront is not always the most energetic arrival.

The Lagrangian solver is used to identify regions in the Marmousi model where sources generate later arrivals for at least one receiver on the surface. This is done by evaluating 2976 source points with a vertical and horizontal spacing of 48 m. Figure 3.18a shows a map of the maximum difference in arrival time between the first and second arrival at the surface (as measured by the receivers), plotted at each source point. Sources close to the receivers in the centre of the model show a large difference in travel time between the first and second arrival. This is due to the strong velocity gradient in the lower left region of the model, which behaves like a reflector for sources above the fast region and deflects down-going energy back towards the surface. Figure 3.18b is a map of the maximum number of later arrivals at the surface, as a function of source location. For reasons of computational convenience the maximum number of arrivals is limited to 60. Sources close to the left and right boundary of the model tend to generate large numbers of later arrivals, as the wavefronts can travel greater distances and therefore more and larger swallowtails can develop. If a source is located in either of the two fast regions (i.e. near to the left and right edges of the model), one does not observe large numbers of later arrivals at the surface. A wavefront in the neighbourhood of a point source tends not to develop a swallowtail, even in the presence of a significant velocity contrast, as a result of its high curvature. Figure 3.18c shows a velocity gradient map of the Marmousi model. By comparing the wavefronts in figure 3.16 with figure 3.18c it can be observed that the swallowtails are initiated in regions where a strong velocity gradient is observed. This example shows that the generation and detection of multiple arrivals is extremely sensitive to small changes in velocity structure, and source and receiver location.

3.2.5 Surface wave multipathing

Two surface wave velocity models are used in order to demonstrate the ability of the Lagrangian solver to compute wavefronts and ray paths for surface waves

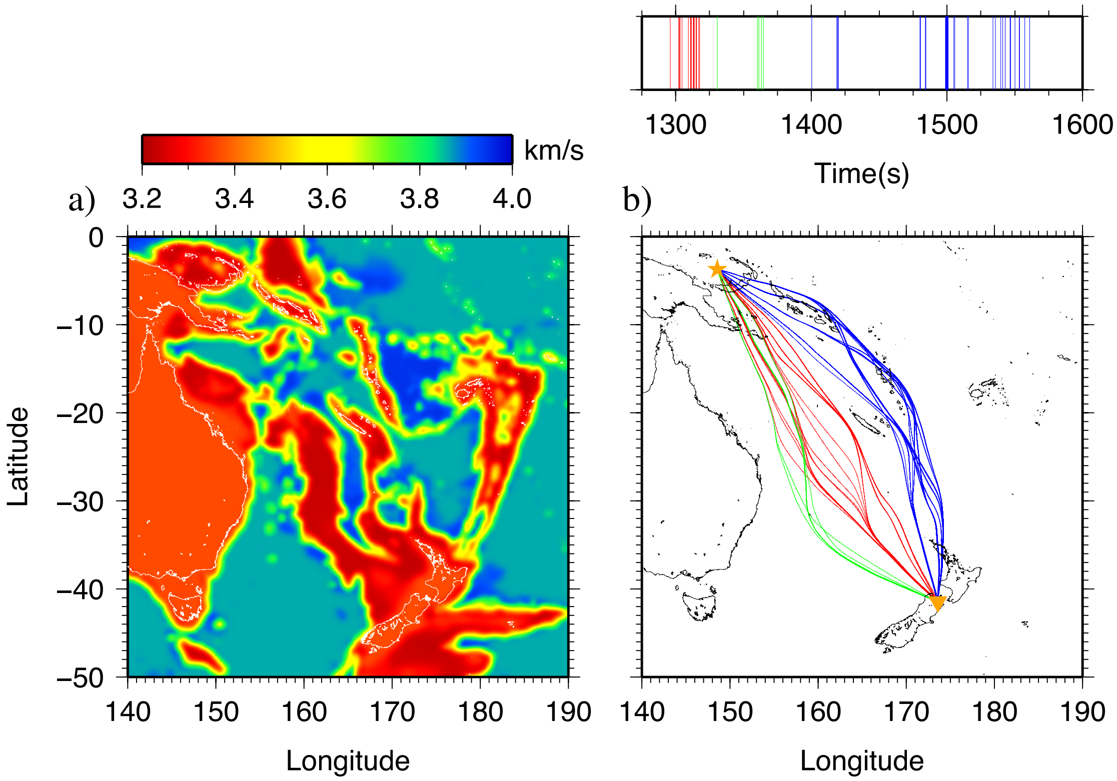


Figure 3.19: (a) Synthetic phase velocity model for fundamental mode Rayleigh waves with a period of 15 s for the eastern Australian region (b) 61 ray paths between source and receiver obtained by the Lagrangian scheme and coloured according to their arrival time. Note how each package of rays samples a different region of the model.

in highly complex structures. The commonly used approximation that surface waves travel along the great circle between source and receiver is only valid at low frequencies (e.g. Jordan, 1978; Dziewonski, 1984; Woodhouse & Dziewonski, 1984). For Rayleigh waves at frequencies greater than about 0.05 Hz, however, the influence of lateral contrasts in the velocity structure becomes significant (e.g. Sobel & von Seggern, 1978). The propagation of higher frequency surface waves in media with strong lateral velocity variations can therefore be approximated by ray tracing or wavefront tracking.

In the first example, a velocity model for fundamental mode Rayleigh waves at a period of 15 s (see figure 3.19a) is used. The velocity model was generated by working with a basis set of five dispersion curves (continental, submerged continental with 1 km of water, submerged continental with 2 km of water, oceanic with 3 km of water, oceanic with 4 km of water) and then interpolating using the average local topography (Sambridge et al., 1993). The resulting structure is not

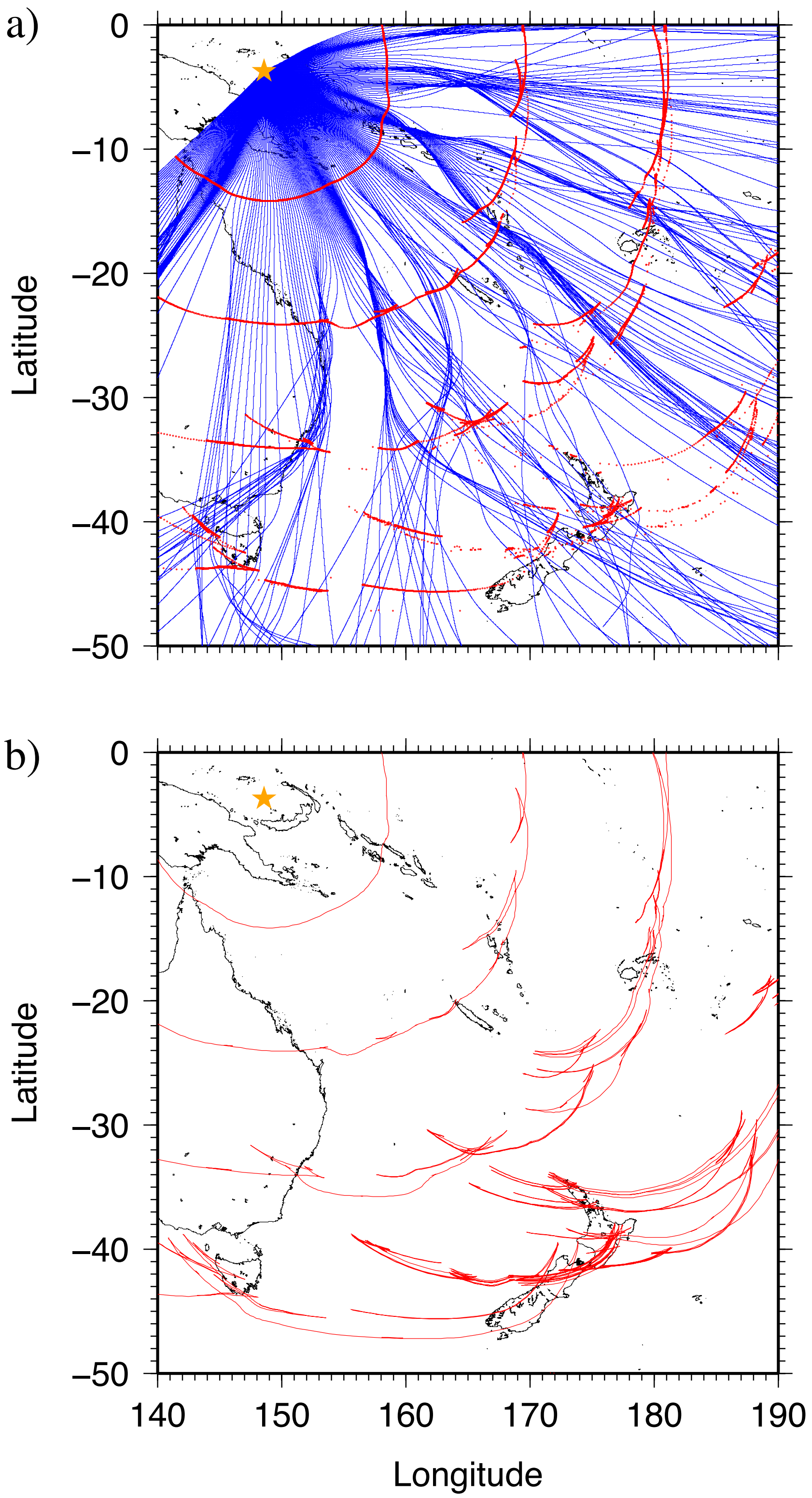


Figure 3.20: Snapshots of wavefronts at 330 s intervals calculated using (a) the shooting method and (b) the Lagrangian wavefront tracker. The red dots in (a) represent the wavefronts computed using the shooting approach. There are more red dots than rays because for clarity not all the rays are plotted. Note how incomplete the wavefronts are when the shooting approach is used.

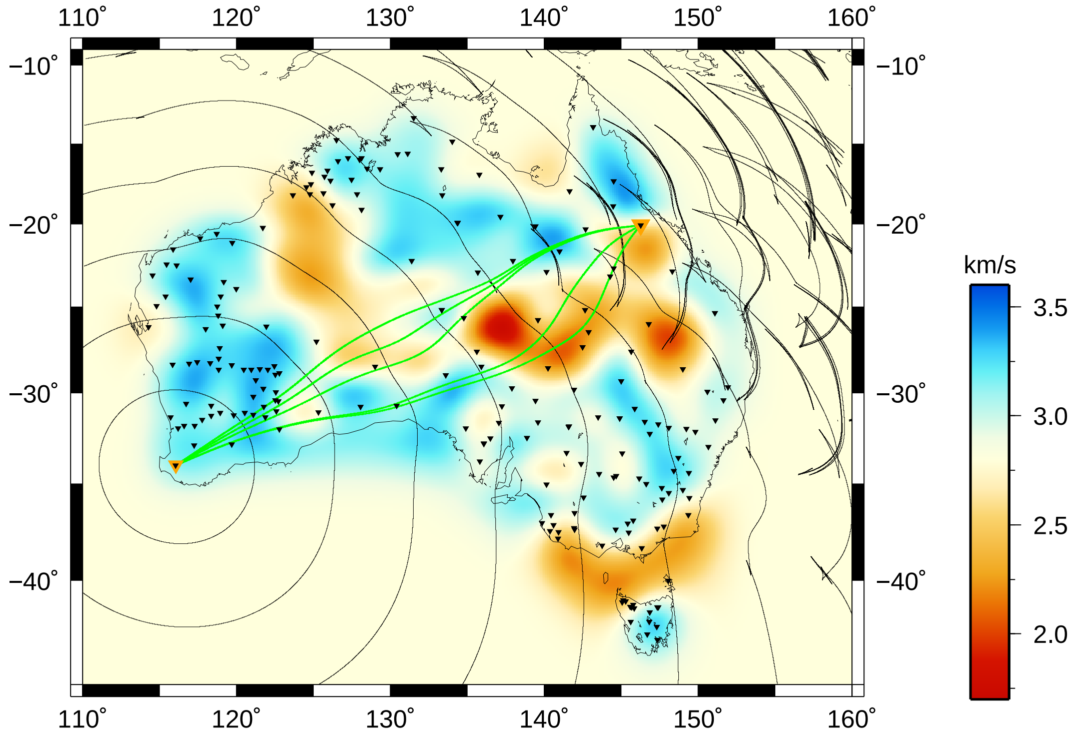


Figure 3.21: Wavefronts at 165 s intervals and ray paths between two receivers in a velocity model for Rayleigh waves with a frequency of 0.2 Hz. The black triangles mark the stations which provided the ambient noise data used to construct the tomographic model (Saygin, 2007). A total of five arrivals can be observed.

intended to be an actual model of surface wave speed, but rather a representative model of typical complexities that one might expect for the region. It has been used before for discussing surface wave tracking in complex structures by Sambridge et al. (1993). The model exhibits strong lateral velocity gradients (see figure 3.19a) which are capable of causing the development of swallowtail patterns.

The Lagrangian wavefront tracker allows a complete understanding to be gained of the multipathing that occurs between a source in New Ireland and receiver on the south island of New Zealand (figure 3.19b). The bicharacteristic strip for the point source is represented by 150 points. The computation time for 3000 iterations with a time step of 0.6 s is 10 s. The maximum number of points used for the representation of the bicharacteristic strip during the tracking of the wavefront is 2971. The scheme can recover 61 arrivals and their corresponding ray paths between source and receiver (Figure 3.19b). Figure 3.20a shows a fan of rays shot from the same source at uniform angular separation. The red points represent the position of the ray endpoints at the same discrete times as the wavefronts in figure 3.20b, which were computed by the Lagrangian wavefront tracker. Due to severe focusing and defocusing effects, the shooting approach cannot fully recover the wavefronts. Rays are bent into and channelled along the low velocity regions and in some cases a significant portion of a wavefront is sampled by only a single ray. The main thing to note is the complexity of the wavefronts in figure 3.20b and the time difference between the first and last arrival. As indicated by the colours in figure 3.19b, rays with similar arrival times tend to sample common regions of the model.

One might argue that the velocity variations in the surface wave model (figure 3.19a) are too large. However, models of Rayleigh wave velocities for Australia obtained by ambient noise tomography show velocity variations of up to 40%, which is sufficient to generate several later arrivals (see figure 3.21). Although the Rayleigh wave velocity model is based on only first arrival information, later arrivals are evident in the correlated noise (Saygin, 2007). Exploiting these later arrivals may significantly improve images of this type.

3.3 Summary

The above examples show that the Lagrangian scheme can be used to track large numbers of later arrivals in complex media and is much more efficient than the Eulerian scheme. On the other hand the Eulerian scheme is based on the level set method, which is a rapidly developing technique. Eulerian schemes using the paraxial wave equation instead of the eikonal equation have been applied to the Marmousi model (Qian & Leung, 2004, 2006) and even been used in a seismic imaging context (Leung & Qian, 2007) albeit in a very limited way. However it is still very much an experimental technique and for the routine computation of multi valued travel times the Lagrangian approach seems to be a much more appropriate method. After all, in the Lagrangian method a two dimensional problem is solved while in the Eulerian scheme two three dimensional problems are solved in order to track the same wavefront.

Tests with a constant velocity gradient model show that the accuracy of the wavefront construction technique is more sensitive to the initial number of points on the bicharacteristic strip than the size of the time step. It was also shown that a higher order interpolation scheme can achieve greater accuracy without sacrificing CPU time, although the linear interpolation scheme is still sufficient for most applications. The robustness of the new scheme is clearly demonstrated by application to the Marmousi model (over 60 later arrivals detected) and surface wave velocity models of the Australian region.

Ray paths associated with the later arrivals clearly sample different parts of the structure compared to the first arrival paths. The potential for exploiting this additional information in seismic travel time tomography will be investigated in chapter 5. So far, only later arrivals that arise from smooth changes in velocity have been computed. Later arrivals can also be observed for reflected and refracted waves due to variations in interface geometry. Therefore in the next chapter interfaces will be added to the Lagrangian scheme.

The identification of later arrivals in recorded seismic wavetrains is a major obstacle to their use in application. The different arrivals of a wavefront will not only have different amplitudes but also different directions when they arrive at a receiver. If one can identify the direction from which the energy of an arrival in a seismogram comes, one can use this to determine which of the predicted later arrivals corresponds to the observed arrival. Relative amplitudes, as computed by dynamic ray tracing for example, may provide some assistance, but ultimately, a synthetic seismogram based on the multipathing information obtained from the Lagrangian solver is likely to be of greatest benefit. In the following chapter a Gaussian beam method is coupled with the Lagrangian solver to achieve this goal.

Chapter 4

Extensions of the Lagrangian scheme

In the previous chapter, the method and applications were directed towards later arrivals produced by smooth changes in wave speed. Discontinuities in the velocity field, for example the boundary of a salt dome, give rise to another class of phase in the form of reflected and refracted wavefronts. These wavefronts may be multi valued, if the impinging wavefront is multi valued, or if the geometry of the interface is such that significant focusing or defocusing occurs. In the following treatment, the Lagrangian scheme is extended so that it can be used to compute multi valued travel times in the presence of interfaces.

In chapter 3, dynamic ray tracing was introduced to compute relative amplitudes; however, travel times and relative amplitudes alone may not be sufficient for the reliable identification of later arrivals. Here the Gaussian beam method is implemented for the computation of ray based synthetic seismograms.

4.1 Interfaces

One of the more popular styles of structural representation in seismic velocity models that include discontinuities is to describe the subsurface by a set of subhorizontal layers. The interfaces between these layers can be parameterized in a variety of ways. For example, in refraction and wide angle reflection tomography linear segments (e.g. Zelt & Smith, 1992; Williamson, 1990), interpolating spline functions (e.g. White, 1989; Lutter & Nowack, 1990) and cubic B-splines featuring local control of interface geometry have been used (e.g. Farra & Madariaga, 1987;

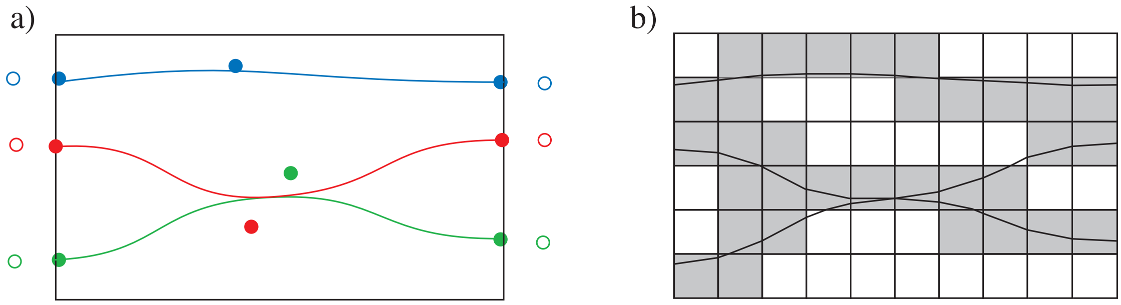


Figure 4.1: (a) Three interfaces (blue, red and green line) are given by a set of control points (blue, red and green points). Each cubic B-spline segment is based on four nodes. Therefore additional nodes outside the model domain are needed, marked using open circles. (b) The interfaces are then sampled onto an interface grid, where they are approximated by a set of linear segments.

Virieux & Farra, 1991; Rawlinson et al., 2001). In more complex structures, layers may pinch-out, interfaces could overturn and isolated bodies may be encountered. However, even a simple non planar interface can cause the generation of later arrivals for reflected and refracted waves irrespective of the velocity field present in adjacent layers. Therefore, later arrivals not only contain additional information about velocity structure, but also about interface geometry.

4.1.1 Representation of an interface

For a heterogeneous model, the continuous wave speed structure has been defined using a regular grid with cubic B-spline approximation in section 3.1. A cubic B-spline approximation provides a smooth representation of the velocity field, and it is desirable to have a similar level of smoothness in the representation of interfaces. An interface in this work is therefore described by a set of control points with cubic B-spline approximation used to describe the position of the interface as a function of incremental path length along the interface (see figure 4.1a). If the interface control nodes are given as $\mathbf{c}_i = (x_i, z_i)$, the position of the interface at an arbitrary path length u along the interface between the nodes i and $i+1$ is given by

$$\mathbf{B}_i(u) = \sum_{l=-1}^{2} b_l \mathbf{c}_{i+l}, \tag{4.1}$$

where $0 \leq u \leq 1$. The weighting factors b_l are given by the uniform cubic B-spline functions (see appendix B). Ghost nodes need to be added beyond the two endpoints of the interface as the cubic B-spline approximation is also dependent on the two neighbouring nodes of the segment (i.e. $\mathbf{c}_i$ and $\mathbf{c}_{i+1}$), within which an interface position is approximated. Cubic B-splines in parametric form have been widely used to describe interfaces in seismological applications (e.g. Farra & Madariaga, 1987; Virieux & Farra, 1991; Rawlinson et al., 2001; Rawlinson & Sambridge, 2004a). Describing the position of an interface as a function of path length along the interface allows overturning boundaries and isolated bodies to be modelled. This would not be possible in a parameterization, where interface depth is defined as a function of horizontal distance.

In the method proposed here, both layers and isolated bodies may be represented. For an isolated body, ghost nodes are not required since $\mathbf{B}_i(u)$ is a periodic function. A regular grid of nodes coupled with cubic B-spline functions is used to describe a smoothly varying velocity field within both layers and isolated bodies. It is usually necessary to extend the description of the velocity field, and hence the grid, beyond the boundaries of the layer or isolated body. These velocity values are redundant unless changes in the interface geometry (e.g. from the application of seismic tomography) cause them to lie within the layers or isolated bodies. The velocity field within each region is designed to be independent of velocity fields in all the other regions. Each layer or isolated body therefore has its own velocity grid which gets activated when the wavefront enters. This concept of assigning an individual velocity grid to each layer can be extended by using two velocity grids for each layer - one for P-waves and one for S-waves, which allows mode conversions.

When tracking a wavefront, one has to verify for each time step whether or not a point on the wavefront has crossed an interface. This requires information about the interface to be stored so that it can be accessed efficiently during wavefront tracking. The approach used in this work is to store information about the interface structure on a regular grid, which in the following will be known as an interface grid (see figure 4.1b). For each cell of the interface grid, the number of interfaces passing through the cell is stored. If a cell contains an interface segment, the position and path length of its start and end point are stored. Within each cell, linear approximation to the ray segment is used. In the subsequent determination of the interface normal at the ray interface intersection point, the cubic B-spline representation of the interface is used. Otherwise, the discontinuities in the direction of the interface normal between adjacent segments could have a destabilising effect on the wavefront propagation. Two ray paths with similar trajectories impinging on either side of a join between two linear interface segments may depart

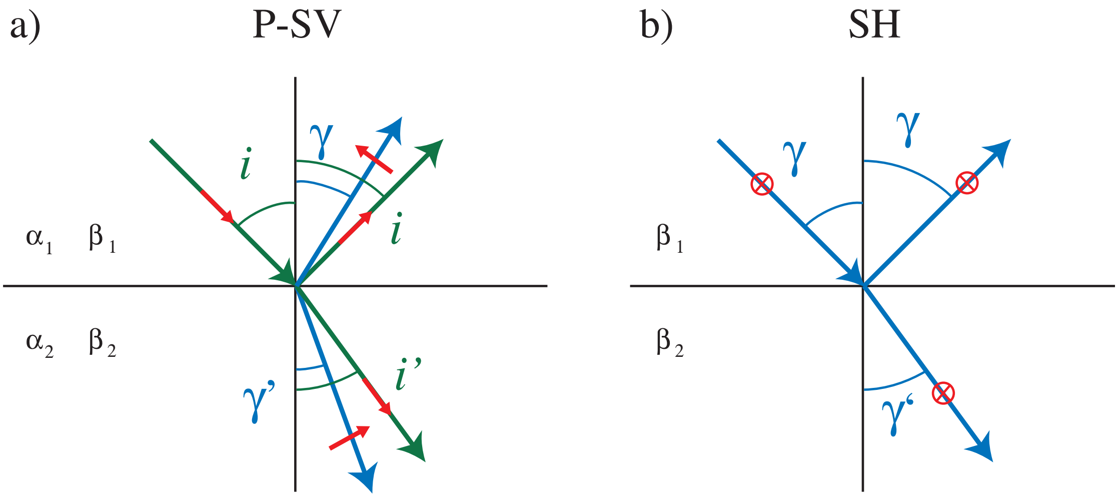


Figure 4.2: Snell's law for *P*-waves (green) and *SH* and *SV*-waves (blue) The direction of the particle motion along each ray is given by the red arrow. Note that for the *SH*-waves the arrow is pointing into the page, because the displacement vectors for *P*, *SH* and *SV*-wave form a right handed coordinate system. (a) Reflection and refraction of an incoming *P*-wave, (b) reflection and refraction of an incoming *SH*-wave.

at very different angles (Zelt & Smith, 1992).

Sampling the interfaces onto the interface grid also allows layer pinch-outs to be readily identified. If in a given cell the distance between two interface segment is smaller than a threshold value the layer between the interfaces is removed and a single new interface is formed. In general, if the grid spacing of the interface grid is small enough, one should be able to adequately represent most interface geometries.

4.1.2 Wavefront propagation in the presence of interfaces

When a seismic body wave encounters a boundary or discontinuity at which the velocity changes, the incoming wave can split into reflected and refracted waves. In some circumstances, a critically refracted or head wave can also be generated. The physics that govern wave propagation require that stress and displacement are continuous across an interface. The refraction of a *P*-wave at an interface causes particle motions that are not parallel on opposite sides of the interface, except for normal incident *P*-waves. Therefore, the *P*-wave displacement alone is not continuous across the interface. An additional particle motion is required to ensure the continuity of displacement across the interface. This additional particle motion is provided by an *SV*-wave in both layers in a direction perpendicular to the displacement of the reflected or refracted P-wave (cf. figure 4.2). In the case of a normal incident P-wave the particle motion is perpendicular to the interface and no SV-wave is generated. For an SH-wave interacting with an interface no additional particle motion is required because the direction of the particle motion is parallel to the interface.

In summary, for an isotropic medium, if a P or SV-wave impinges on an interface, four derivative waves can be generated: a reflected P and SV-wave and a refracted P and SV-wave (figure 4.2a), while for an SH-wave, only two derivative waves result, a reflected and a refracted SH-wave (figure 4.2b). The ray trajectories of refracted and reflected waves is governed by Snell's law. The waves must move along the interface with the same apparent velocity and therefore

$$\frac{\sin i}{\alpha_1} = \frac{\sin \gamma}{\beta_1} = \frac{\sin \gamma'}{\beta_2} = \frac{\sin i'}{\alpha_2}, \tag{4.2}$$

where α_1 and α_2 are P-wave velocities in the two layers and β_1 and β_2 are corresponding S-wave velocities. The angles of inclination of the P-wavepaths in layer 1 and 2 are given by i and i' and for the S-wavepaths by γ and γ'. The critical angle i_c above which no energy can penetrate into the second layer is given by

$$i_c = \sin^{-1}\left(\frac{v_1}{v_2}\right), \tag{4.3}$$

where v_1 and v_2 are the wave speed in the two layers for the corresponding waves. If the angle of incidence is equal to the critical angle a critical refracted wave or head wave is generated, which will travel along the interface with the velocity v_2. While it would be possible to compute wavefronts for critical refracted waves, the Lagrangian solver developed in this work is not designed to do so.

The Lagrangian scheme presented in chapter 3 is intended for a continuous velocity field. If there are multiple layers present, a natural approach for tracking the wavefront is to consider its propagation in each layer separately. Such a multi stage method for computing travel times in a layered model has also been used by Rawlinson & Sambridge (2004b), in the context of implicit wavefront tracking using the fast marching method. Figure 4.3 illustrates this multi stage approach.

In a given layer, a wavefront is propagated until all points on the bicharacteristic strip have either left the computational domain or hit an interface. When the position of a point on the bicharacteristic strip is updated for a given time step, a check is made to see whether a portion of the path between its current and

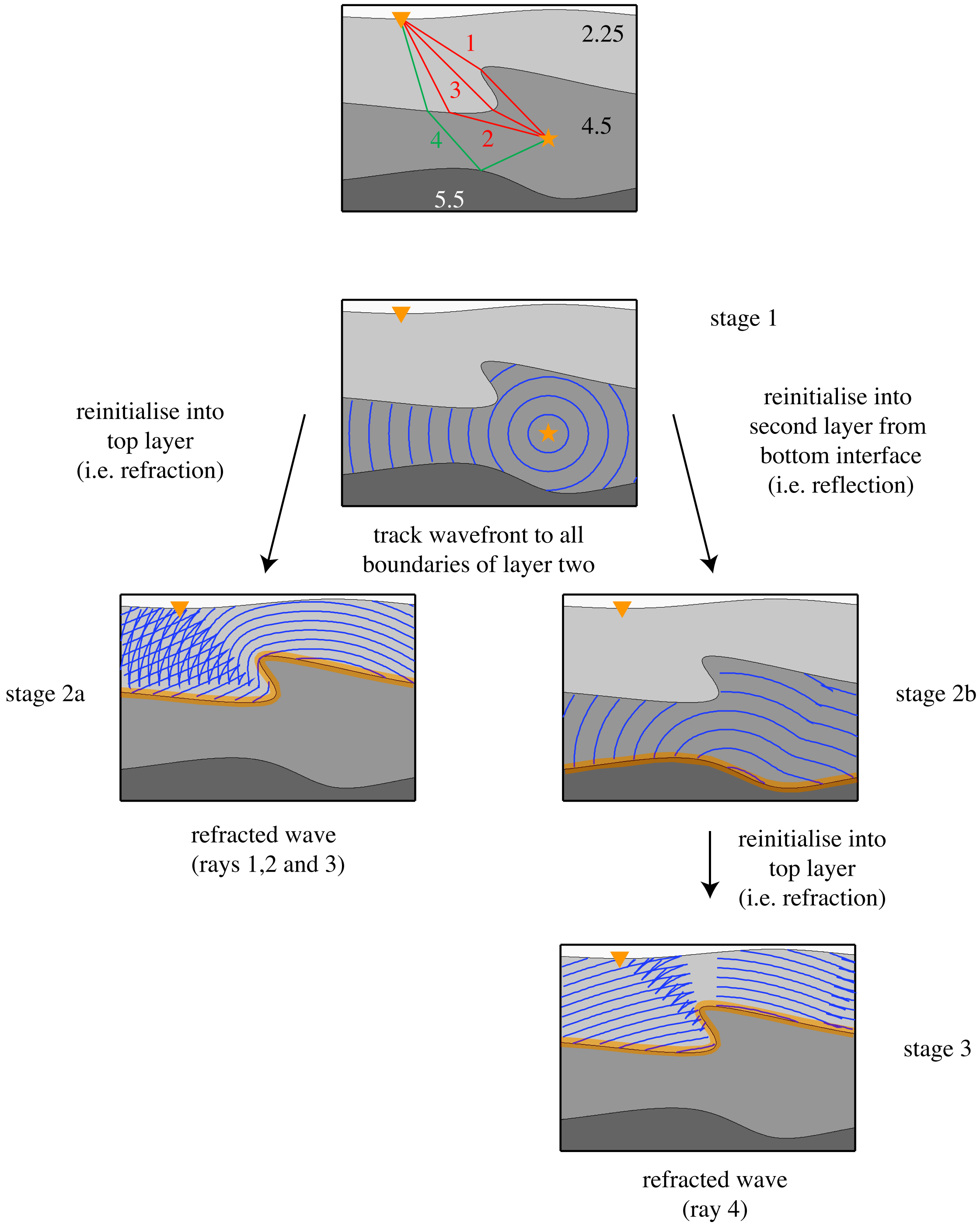


Figure 4.3: The steps of a multi stage approach for the computation of ray paths for a direct wave (red rays) and a wave reflected from the bottom interface (green ray). Velocities for the three layers are given in the top diagram. Both path signatures share the wavefront emanating from the source in the middle layer. Note how the reflected and refracted wavefronts triplicate due to the shape of the interfaces and how the triplications can be propagated across interfaces. The interfaces from which the wavefront starts are highlighted in orange. The wavefront computed in stage 3 shows a gap due to the overturning middle interface, which splits the upcoming wavefront.

previous position lies in a grid cell which contains an interface. If this is the case, the next step is to test whether the path segment intersects one or more interface segments inside the cell. If more than one intersection point is identified, the one closest to the starting point of the ray segment is chosen. This point, which now lies on the updated bicharacteristic strip, is no longer updated during future time steps for the current wavefront. The position of the point on the interface, its angle of incidence, and the exact time at which it has hit the interface are stored.

Having finished the propagation of a wavefront in a given layer, all the points have either left the computational domain or are lying on an interface. In the latter case the arrival time, angle of incidence and local direction of the interface normal are known, so a reflected or refracted wave can be initialised. Following application of Snell's law, the starting position and time of each point of the departing bicharacteristic strip is now known. The initialisation time of the new wavefront is set to the time at which the incident wavefront first impinges on the interface. The bicharacteristic strip is then propagated as described earlier and points are added to the set of points on the evolving wavefront once the time at which they originally hit the interface is reached.

While it is possible to calculate all wavefronts generated by the interaction of an incident wavefront with an interface, one has to keep in mind that at each interface four new wavefronts are generated (if P and SV-waves are considered), which eventually yields, in complex media, a large number of wavefronts. It is therefore more convenient to only compute the paths which are needed for the problem at hand. In this scheme a path is defined by a set of segments, where each segment contains information about the wavefront to be tracked in a given layer i.e. the origin of the wavefront (source point or an interface) the type of wave (P, SV or SH and direct, reflected or refracted) and its destination (an interface identifier). In conventional ray tracing methods this type of phase identification is often referred to as a path signature or a ray code (e.g. Červený, 2001). A path signature basically tells the wavefront tracking scheme which wavefronts have to be propagated in which layers and in what sequence. For any path signature the new scheme computes first and later arrivals if they exist. This means that a path signature may represent a family of rays between source and receiver, as illustrated in figure 4.3 where the three red rays have the same path signature but different arrival times and propagation paths.

When several path signatures are specified for a particular source, the paths often differ from each other only after a number of reinitialisation steps. Every path will, for example, require the wavefront which propagates through the region in which the source is located. In order to avoid having to recompute wavefronts, a tree structure containing all the wavefronts needed for the different path signatures is built (see figure 4.3). The first node of the tree is the wavefront which propagates through the region in which the source is located. The children of that node are then given by wavefronts generated as a result of interactions between the initial wavefront and adjacent interfaces as required for the different phases. In the scheme presented in this work a tree is built so that it contains all required paths. The algorithm then begins at the root of the tree and visits all nodes of the tree so that only the required wavefronts are propagated. As shown in figure 4.3 the wavefront for certain path signatures may have gaps. In this case, not every point on the surface can be reached by a ray which bounces off the lower boundary and then refracts at the upper boundary of the middle layer (see figure 4.3 stage 2b). Therefore the wavefront associated with this phase develops a gap, as shown in stage 3.

For a specified arrival at a receiver, the corresponding ray path signature must be retained so that a ray path can be constructed a posteriori. Compared with the situation in chapter 3 more than one wavefront is now tracked. Consequently it is no longer feasible to keep all wavefronts in computer memory for the back tracking of a ray once the wavefront tracking is finished. Therefore, the wavefronts for each time step are stored in a scratch file on disk in the version of the scheme used here. A different tree structure is then used so that a single wavefront can be read directly, without having to search the whole scratch file.

4.1.3 Ray path extraction in the presence of interfaces

The process of back tracking a ray from the receiver to the source once the wavefront propagation is finished has been described in section 3.1.2. This concept can be extended to a model containing interfaces. In order to back track across interfaces, one needs to know the origin of the wavefront. It may emanate from the source, or be a reflection or refraction from an interface. This information can be obtained from the tree structure used to propagate the different wavefronts.

In figure 4.4 a plane wave travels through layer 2, impinges on an interface, and generates a refracted wave. This means that two separate wavefronts have been computed, and a ray path has to be back tracked across the interface from layer 1 into layer 2. For a given time a point on the ray path is either on the wavefront in

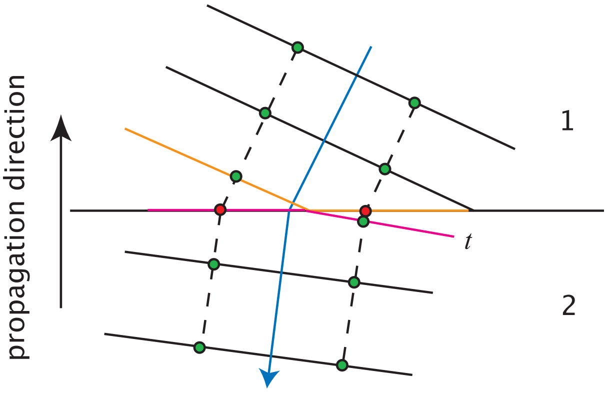


Figure 4.4: Back tracking a ray path for a plane wave crossing an interface from layer two into layer one. The ray path to be extracted is given by the blue line and the points used to describe the wavefronts are given by the green and red dots.

layer 1 or layer 2. Hence, one needs to verify if the ray path segment is going to intersect the interface for the current time step. Once the ray path has intersected the interface, the wavefront in the adjoining layer must be used.

It is possible that the point on the wavefront on either side of the current ray path is not available for interpolation because it is attached to the interface for the current time i.e. this part of the wavefront is yet to be initialised in the current layer. This is illustrated in figure 4.4 where the wavefronts in the two layers at time t are given by the orange and magenta line. The red points on the interface belong to the corresponding wavefront, but are not at a valid position for the current time. When this occurs, the absolute distance to the nearest green point is used, instead of the relative distance between adjacent green points. Once both neighbours are again available, the standard approach can be used (see section 3.1.2). In order to avoid complications caused by adding and removing points on a wavefront near the interface, the wavefront tracking scheme has to be modified. A point is now only added or removed if both neighbouring nodes are not sitting on the interface at the current time.

Similarly, if the ray path is to be extracted for a reflection, we must make sure that only points which are at a valid position for the current time step are used for the construction of the ray path. The approach described here is capable of extracting a variety of ray paths for complex models, as shown in figure 4.5. In this example, ray paths of multiply reflected and refracted waves are extracted for a structure which contains an overturning interface, a layer pinch-out and an isolated body.

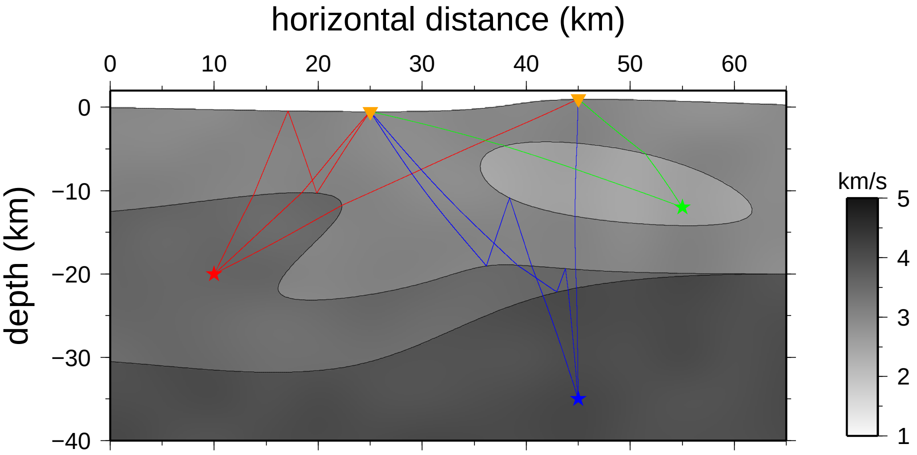


Figure 4.5: Complex model with a layer pinch-out, an isolated body and an overturning interface. For three separate sources, ray paths with different path signatures are given in red, blue and green.

4.2 Gaussian beam method

The Gaussian beam method (Popov, 1982; Červený & Pšenčík, 1984) is an asymptotic approach for the computation of seismograms in homogeneous and inhomogeneous media based on a combination of geometric ray concepts and elements of wave theory. Since high frequency energy is considered to propagate along a ray, one can for a given ray solve the wave equation in ray centred coordinates. A parabolic approximation can be used to find the asymptotic local solution in the neighbourhood of each ray. The wavefield at a receiver is then given by a superposition of the displacement field computed for the family of rays that pass near the receiver.

Stacy & Nowack (2002) use the Gaussian beam method in two dimensions to model seismic attributes in a wide angle refraction study. In the field of reflection seismology Hill (2001) discusses a Gaussian beam migration method, which is based on reversing the steps of Gaussian beam forward modelling. Dunn & Forsyth (2003) use a Gaussian beam method to model observed packets of Love waves. Finally, it has also been used for modelling teleseismic P-waves in three dimensional structures (Cormier, 1987).

Combining the Gaussian beam method with wavefront tracking will for the first time allow narrow fans of rays to be shot towards the receiver without the risk of computing synthetic seismograms which do not contain all arrivals.

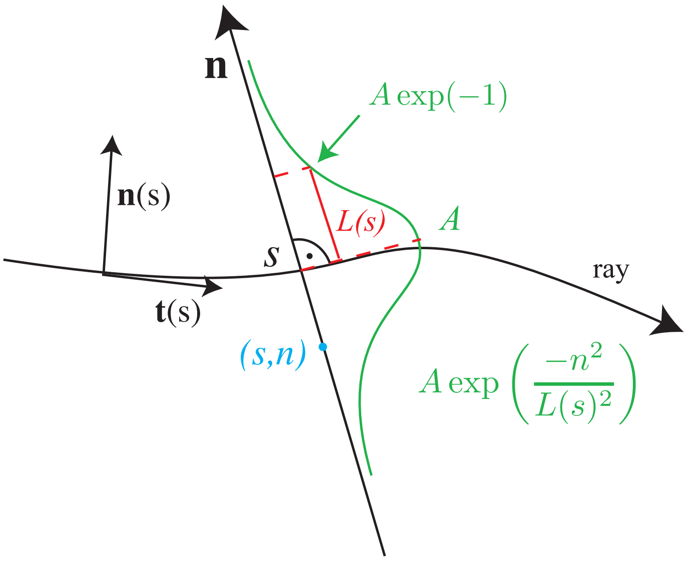


Figure 4.6: Ray centred coordinates s and n for a two dimensional medium; the amplitude profile (green curve) along $\mathbf{n}$ for a Gaussian beam is also shown. A is the maximum amplitude and $L(s)$ is the effective half width of the beam (red line) (cf. Červený et al. (1982)).

4.2.1 Gaussian beams in smoothly varying media

The wave equation in two dimensions (Aki & Richards, 2002) is given by

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial z^2} = \frac{1}{v^2}\frac{\partial^2 u}{\partial t^2}, \tag{4.4}$$

where v is velocity, t is time, x and z are the Cartesian coordinates, and $u(x,z,t)$ represents the displacement field.

The basic idea behind the Gaussian beam method is to express the displacement field $u(x,z,t)$ in the vicinity of a ray using a ray centred coordinate system. Figure 4.6 shows a ray centred coordinate system for a path in a two dimensional medium. The coordinate s measures the arclength along the ray from an arbitrary reference point and n represents a length coordinate in the direction perpendicular to the ray given by the vector $\mathbf{n}(s)$.

Červený et al. (1982) use the parabolic wave equation method to derive an expression for the displacement field $\mathbf{u}(s,n,\omega)$ near a ray in the angular frequency domain using ray centred coordinates:

$$\mathbf{u}(s,n,\omega) = A(s)\exp\left(-i\omega\tau(s) + \frac{i\omega}{2v(s)}K(s)n^2 - \frac{n^2}{L(s)^2}\right)\mathbf{a}(s), \tag{4.5}$$

with $K(s)$ and $L(s)$ given by

$$K(s) = v(s)\mathrm{Re}\left(\frac{p(s)}{q(s)}\right), \tag{4.6}$$

$$L(s) = \left[\frac{\omega}{2}\mathrm{Im}\left(\frac{p(s)}{q(s)}\right)\right]^{-\frac{1}{2}}. \tag{4.7}$$

$A(s)$ is the complex amplitude, $\tau(s)$ is the travel time along the ray to the coordinate s, $K(s)$ is the wavefront curvature, $L(s)$ the frequency dependent effective half width of the beam and ω the angular frequency. The frequency dependent effective half width of the beam is the distance from the central ray along $\mathbf{n}$ at which the amplitude of the Gaussian beam is $1/e$ times the amplitude of the central ray A. While $L(s)$ is in fact only the half width of the beam (Červený et al., 1982) it is commonly referred to as beam width (e.g. Nowack & Aki, 1984; Weber, 1988). The direction of the particle motion is given by $\mathbf{a}(s)$, which is equal to $\mathbf{t}(s)$ for a P-wave, $\mathbf{n}(s)$ for an SV-wave and $\mathbf{t} \times \mathbf{n}$ for an SH-wave.

The exponential decrease of amplitude with increasing distance from the central ray (i.e. n) in (4.5) is Gaussian and therefore the amplitude profile along $\mathbf{n}$ is bell shaped. Consequently the solution (4.5) of the parabolic wave equation is called a Gaussian beam. To construct the wavefield at a given receiver, contributions from the Gaussian beams travelling in the vicinity of the receiver are superpositioned.

Dynamic ray tracing (see section 3.1.3) plays a crucial role in the computation of Gaussian beams. It is used to determine the quantities $q(s)$ and $p(s)$ in (4.5). Replacing the time t with the path length along the ray s using

$$\frac{\mathrm{d}}{\mathrm{d}s} = \frac{\mathrm{d}}{\mathrm{d}t}\frac{\mathrm{d}t}{\mathrm{d}s} = \frac{1}{v}\frac{\mathrm{d}}{\mathrm{d}t} \tag{4.8}$$

the dynamic ray tracing equations (3.5) can be written in matrix form as

$$\frac{\mathrm{d}}{\mathrm{d}s}\begin{bmatrix} q \\ p \end{bmatrix} = \begin{bmatrix} 0 & v \\ -v^{-2}v_{\mathbf{nn}} & 0 \end{bmatrix}\begin{bmatrix} q \\ p \end{bmatrix}, \tag{4.9}$$

where $v_{\mathbf{nn}}$ is the second derivative of the velocity v in the direction of the ray perpendicular $\mathbf{n}$. For Gaussian beams, Červený et al. (1982) define $p(s)$ and $q(s)$

as

$$q(s) = \varepsilon q_1(s) + q_2(s),$$
$$p(s) = \varepsilon p_1(s) + p_2(s), \tag{4.10}$$

where ε is a complex valued parameter which will be determined later. The fundamental matrix of linearly independent real solutions of the system (4.9) is

$$\begin{bmatrix} q_1 & q_2 \\ p_1 & p_2 \end{bmatrix}. \tag{4.11}$$

The corresponding initial conditions are given by

$$\begin{bmatrix} 1 & 0 \\ 0 & 1/v_0 \end{bmatrix}, \tag{4.12}$$

where v_0 is the wave speed at the initial location of the ray. The Gaussian beam method is discussed here for a two dimensional structure and out of plane spreading is not included. The two columns of (4.12) are the initial conditions for two line sources perpendicular to each other (e.g. Červený et al., 1982; Nowack & Aki, 1984). The first column represents a line source located in the plane (i.e. it appears in the plane of the model as a plane wave at the source) and the second column corresponds to a line source perpendicular to the plane (i.e. it appears in the plane of the model as a point source). The complex solution for $q(s)$ and $p(s)$ is a linear combination of the real solution for a plane wave and line source weighted by ε (4.10). From these initial conditions it follows that the wavefront curvature of the initial plane wave is

$$K_1(s_0) = v(s_0)\frac{p_1(s_0)}{q_1(s_0)} = 0 \tag{4.13}$$

and for the line source

$$K_2(s_0) = v(s_0)\frac{p_2(s_0)}{q_2(s_0)} = \infty, \tag{4.14}$$

where $s_0 = 0$ is the arclength along the ray of the source location. One would expect these results for the curvature of the two line sources mapped on to the plane of the model, as a line has a curvature of zero and a circle with an infinitesimally small radius has an infinite curvature.

Recall that in dynamic ray tracing $q(t)$ is real and becomes 0 at caustics which leads to a singularity in amplitude (see section 3.1.3). In the Gaussian beam method the factor ε is chosen in such a way that (a) no singularities in amplitude can occur along the ray i.e. $q(s) \neq 0$; and (b) the solution is concentrated near the ray i.e. $\mathrm{Im}\left(\frac{p(s)}{q(s)}\right) > 0$. Červený et al. (1982) therefore write ε in the form

$$\varepsilon = S_0 - i\frac{\omega}{2v(S_0)}L_M^2, \tag{4.15}$$

where geometrically $L_M = L(s = S_0)$ is the beam width at a specific location along the ray referred to as the beam waist S_0, and i is the imaginary unit. The beam waist S_0 is a user defined parameter used to shift the position where the beam width L_M is defined along the ray. Červený et al. (1982) show that conditions (a) and (b) are satisfied when $L_M \neq 0$. This result is based on the determinant of the fundamental matrix (4.11) being a nonzero constant along the ray. Therefore $q_1(s)$ and $q_2(s)$ cannot simultaneously be zero for a given s. Hence there can be no singularities in the amplitude, even in the presence of caustics. This is advantageous when compared to standard dynamic ray tracing (see section 3.1.3) where caustics can lead to singularities in the amplitude field.

The complex amplitude $A(s)$ is computed using (3.7) with the difference that q and p are complex and functions of s, where s is the path length along the ray for which the corresponding travel time is

$$t = \int_0^s \frac{\mathrm{d}s}{v(s)}. \tag{4.16}$$

It is often convenient to define a frequency independent parameter L_0, which specifies the initial beam width at the beam waist. The beam width L_M at the beam waist S_0 is then given by $L_M = (2v(S_0)/\omega)^{1/2}L_0$.

In order to estimate the displacement field at an arbitrary receiver, a wide fan of rays is usually traced from the source (e.g. Červený et al., 1982; Nowack & Aki, 1984; Weber, 1988). The different rays are then identified by their ray parameter. For a point source, the ray parameter can be the takeoff angle, and for an initial plane wave, it can be the distance along the wavefront. In the angular frequency domain, the total displacement at a receiver is given as the integral over all contributions of the Gaussian beams to the displacement field at the receiver. The point along a ray at which the contribution to the displacement field is computed

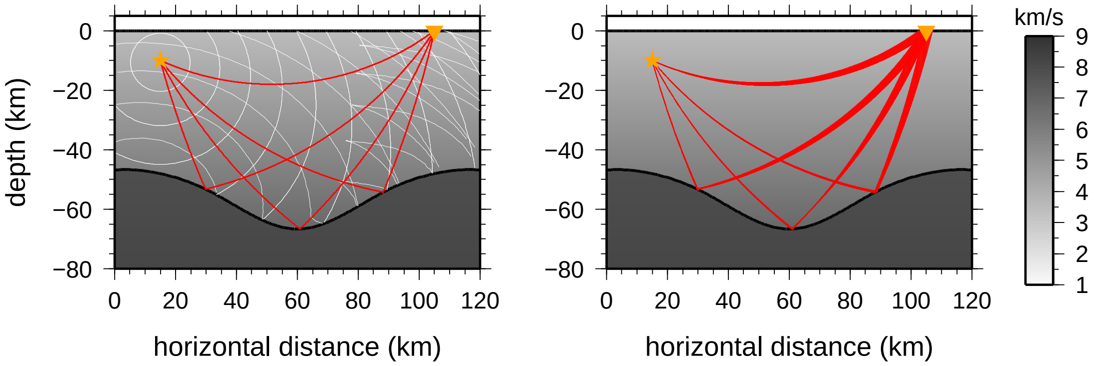


Figure 4.7: Ray path information obtained by the wavefront tracking (left) is used in the Gaussian beam method to shoot only those rays that hit the surface in the vicinity of the receiver (right). Note that for illustration purposes, the ray fans are wider than those actually used for the computation of a synthetic seismogram.

can be either its intersection with the surface (e.g. Nowack & Aki, 1984) or an integration axis, which contains the receiver location and is perpendicular to the ray that intersects the receiver (e.g. Weber, 1988). Typically, the initial conditions of rays that hit the surface close to the receiver are not known, so the fan of rays shot from the source tends to be quite wide, which can lead to considerable redundancy (section 1.2).

The wavefront tracking approach used in this work allows the width of the ray fans to be precisely controlled (see figure 4.7). Having extracted the ray paths for each arrival at a receiver, we know the ray parameters of rays hitting the surface close to the receiver. Since it is only these rays that contribute to the displacement field, one can shoot a narrow fan of rays, and therefore avoid solving the dynamic and kinematic ray tracing equations for redundant rays. In this work, for each source-receiver ray path, a fan of 15 rays is usually shot (i.e. 7 rays on either side of the source-receiver ray path). The rays are evenly distributed so that the beam width of the central ray at the surface is spanned with rays. This will increase the efficiency of the procedure, since the wavefront tracking only requires the solution of the kinematic ray tracing equation for a relatively small number of nodes on the bicharacteristic strip.

The final step of the Gaussian beam method is a superposition of the beam solutions for rays in the vicinity of a receiver. The integral for the two dimensional case is given by

$$\mathbf{u}(\omega) = F(\omega) \int \phi(\vartheta, \omega) \mathbf{u}_\vartheta(s, \omega)\, \mathrm{d}\vartheta, \tag{4.17}$$

where $F(\omega)$ is the source spectrum, $\mathbf{u}_\vartheta(s, \omega)$ the contribution to the displacement

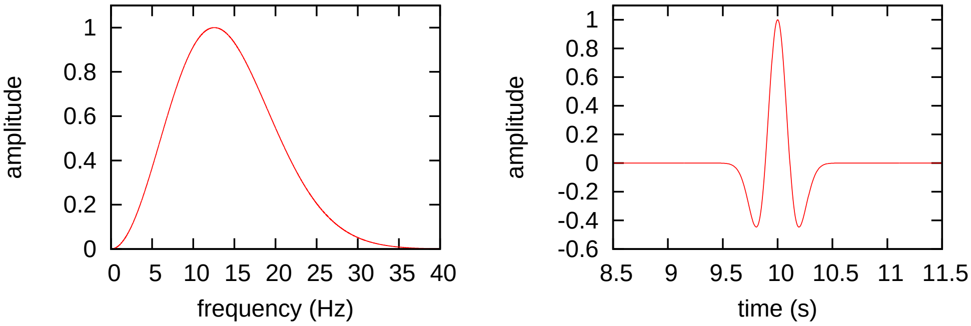


Figure 4.8: Amplitude spectrum (left) and time series (right) of a Ricker wavelet with $\omega_{peak} = 2\,\mathrm{rad} \cdot \mathrm{s}^{-1}$ (i.e. $\nu_{peak} = 4\pi\,\mathrm{Hz}$) and $t_i = 10\,\mathrm{s}$.

by the ray with ray parameter ϑ and $\phi(\vartheta, \omega)$ a weight function. In practice the integral is approximated by a finite sum over the rays hitting the surface in the neighbourhood of the receiver.

A Ricker wavelet (see figure 4.8) can be used for the source spectrum $F(\omega)$ (e.g. Červený, 2001). In the angular frequency domain, a Ricker wavelet is given by

$$F(\omega) = 4\omega^2 \frac{\sqrt{\pi}}{\omega_{peak}^3} \exp\left(-\left(\frac{\omega}{\omega_{peak}} \right)^2 + it_i\omega \right), \tag{4.18}$$

where ω_{peak} is the peak angular frequency and t_i defines the position of the centre of the wavelet on the time axis. If the convolution with the source spectrum is not performed, one obtains the impulse response of the structure instead of a synthetic seismogram.

The three components of the synthetic seismogram (P, SH and SV) in the time domain are given by the inverse Fourier transform of each component of $\mathbf{u}(\omega)$:

$$\mathbf{u}(t) = \mathcal{F}[\mathbf{u}(\omega)] = \frac{1}{2\pi} \int_0^\infty \mathbf{u}(\omega) e^{i\omega t}\,\mathrm{d}\omega. \tag{4.19}$$

The resulting displacement field at a receiver is influenced by four free parameters chosen by the user: the width of the fan of rays used to describe the displacement field, the number of rays forming the fan, the initial beam width L_0 at the beam waist and the shift of the beam waist away from the source given by the parameter S_0. The values chosen for these parameters can influence the shape of the resulting waveform significantly and interference between the Gaussian beam solutions for the different rays (i.e. the contribution of each ray to the final displacement) can distort the resulting seismograms due to constructive and destructive interference.

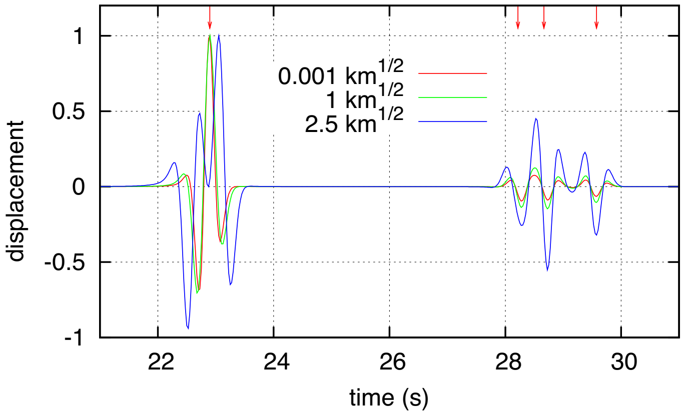


Figure 4.9: Vertical component of synthetic seismograms obtained by the Gaussian beam method for the example in figure 4.7 using different initial beam widths L_0. The seismograms have been scaled so that the absolute maximum amplitude is one. The first arrival corresponds to a direct wave. The second wave package is formed by the wavefront which is reflected from the interface and has triplicated due to the shape of the interface. The red arrows mark the expected arrival times. If the beam is too broad, interference between the rays distorts the first arrival.

A common choice for the beam waist S_0 is to locate it at the source for a point source (i.e. $S_0 = 0$) and at the intersection point of the ray with the surface for a plane wave. This approach to choosing S_0 has previously been used by Nowack & Aki (1984). Figure 4.9 illustrates the effect the beam width has on the resulting seismograms for the scenario in figure 4.7. Clearly if the initial beam width is too large, the interference between the different rays leads in this example to a situation where the shape of the first arrival is distorted.

Knowing the ray parameter of the ray that hits the receiver allows us to use relatively narrow Gaussian beams with sufficient rays in the vicinity of the receiver to contribute to the displacement field. This helps to avoid interference between the rays, but it still allows a waveform at the receiver to be constructed based on more than just the information along the kinematic ray path between source and receiver, which is desirable considering that only the high frequency part of the energy follows the kinematic ray path.

The method can be extended further to consider attenuation when computing the synthetic seismograms. Weber (1988) represents absorption by a power law for the quality factor and includes this in his formulation of the Gaussian beam method.

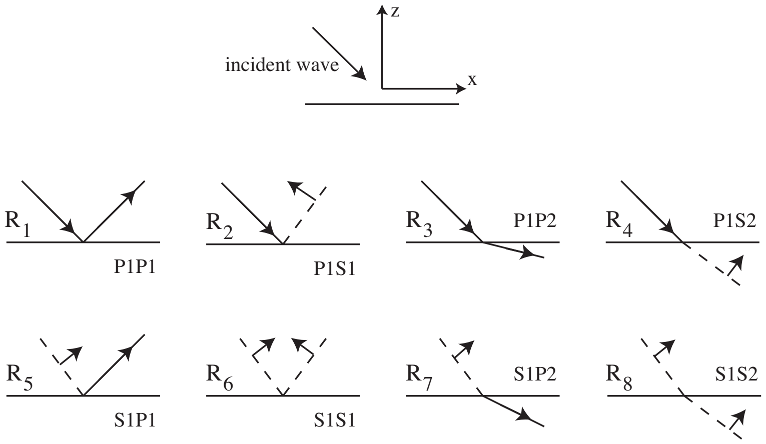


Figure 4.10: Reflection and refraction P-SV coefficients. The arrows point in the positive direction of the displacement vector.

4.2.2 Reflection and refraction of Gaussian beams

Although Snell's law and ray theory can be used to predict the trajectories of wave energy at an interface, one must return to a wavefield representation to determine the amplitude partitioning at an interface. Reflection and refraction coefficients for the displacement vectors at an interface can be derived using plane wave potentials for the various wave components (e.g. Lay & Wallace, 1995; Aki & Richards, 2002). The reflection and refraction coefficients in this work are the so called displacement coefficients (cf. Červený et al., 1977; Červený & Pšenčík, 1984), which give the ratio of a component of the displacement vector of a reflected or refracted wave to any component of the displacement vector of the incident wave (see figure 4.10). They depend on the angle of incidence and on the velocities and densities on both sides of the interface at the point of incidence. Instead of the angle of incidence ϑ, the ray parameter P is used, where v is the wave speed for the incoming ray at the point of intersection,

$$P = \frac{\sin\vartheta}{v}. \tag{4.20}$$

The eight P-SV reflection and refraction coefficients are shown in figure 4.10. In two dimensions, the vector for the particle motion of the SV-wave maintains its orientation with respect to the direction of the P-wave particle motion for an arbitrary number of reflections and refractions at interfaces of any shape. This is due to the displacement vectors for the P, SH and SV-wave forming a right handed coordinate system. The coefficients $R_{1,\dots,8}$ are given by the following equations

(Červený et al., 1977; Červený & Pšenčík, 1984),

$$\begin{aligned}
R_1 =& D^{-1}(Q^2P^2P_1P_2P_3P_4 + \rho_1\rho_2(\beta_1\alpha_2P_1P_4 - \alpha_1\beta_2P_2P_3) \\
&- \alpha_1\beta_1P_3P_4Y^2 + \alpha_2\beta_2P_1P_2X^2 - \alpha_1\alpha_2\beta_1\beta_2P^2Z^2), \\
R_2 =& 2\alpha_1PP_1D^{-1}(QP_3P_4Y + \alpha_2\beta_2XZ), \\
R_3 =& 2\alpha_1\rho_1P_1D^{-1}(\beta_2P_2X + \beta_1P_4Y), \\
R_4 =& -2\alpha_1\rho_1PP_1(QP_2P_3 - \beta_1\alpha_2Z), \\
R_5 =& -2\beta_1PP_2D^{-1}(QP_3P_4Y + \alpha_2\beta_2XZ), \\
R_6 =& D^{-1}(Q^2P^2P_1P_2P_3P_4 + \rho_1\rho_2(\alpha_1\beta_2P_2P_3 - \beta_1\alpha_2P_1P_4) \\
&- \alpha_1\beta_1P_3P_4Y^2 + \alpha_2\beta_2P_1P_2X^2 - \alpha_1\alpha_2\beta_1\beta_2P^2Z^2), \\
R_7 =& 2\beta_1\rho_1PP_2D^{-1}(QP_1P_4 - \alpha_1\beta_2Z), \\
R_8 =& 2\beta_1\rho_1P_2D^{-1}(\alpha_1P_3Y + \alpha_2P_1X), \qquad (4.21)
\end{aligned}$$

where P-wave velocities in the first and second layer are given by α_1 and α_2, the S-wave velocities by β_1 and β_2 and densities by ρ_1 and ρ_2. The parameter D is given by

$$\begin{aligned}
D =& Q^2P^2P_1P_2P_3P_4 + \rho_1\rho_2(\beta_1\alpha_2P_1P_4 + \alpha_1\beta_2P_2P_3) \\
&+ \alpha_1\beta_1P_3P_4Y^2 + \alpha_2\beta_2P_1P_2X^2 + \alpha_1\alpha_2\beta_1\beta_2P^2Z^2,
\end{aligned}$$

and

$$\begin{aligned}
&Q = 2(\rho_2\beta_2^2 - \rho_1\beta_1^2), \quad X = \rho_2 - QP^2, \quad Y = \rho_1 + QP^2, \quad Z = \rho_2 - \rho_1 - QP^2, \\
&P_1 = \sqrt{1-\alpha_1^2P^2}, \quad P_2 = \sqrt{1-\beta_1^2P^2}, \quad P_3 = \sqrt{1-\alpha_2^2P^2}, \quad P_4 = \sqrt{1-\beta_2^2P^2}.
\end{aligned}$$

For $P > 1/v_j$ the ray parameters are given by $P_j = i\sqrt{v_j^2p^2 - 1}$, where $v_1 = \alpha_1$, $v_2 = \beta_1$, $v_3 = \alpha_2$, $v_4 = \beta_2$ and i is the imaginary unit. When the ray impinges on the interface in the negative direction of the x-axis, the sign of the coefficients for the displacement of the resulting SV-waves (i.e. R_2, R_4, R_5 and R_7) must be changed. This is again due to the displacement vectors for the P, SH and SV-wave forming a right handed coordinate system. If the angle of incidence equals the critical angle, a head wave or critical refracted wave is generated. If the angle of incidence is larger than the critical angle, no energy can propagate into the lower layer and the reflected wave experiences a phase shift and the reflection coefficients for this situation become complex. A more detailed discussion of reflection and refraction coefficients is given in Lay & Wallace (1995).

For SH-waves, the displacement vector is parallel to the interface. This means that there is no phase conversion for an incoming SH-wave and only two waves are generated, a reflected SH and a refracted SH-wave. The reflection coefficient R_1 and the refraction coefficient R_2 are defined as follows (Červený et al., 1977)

$$\begin{aligned} R_1 &= D^{-1}(\rho_1\beta_1 P_2 - \rho_2\beta_2 P_4), \\ R_2 &= 2\rho_1\beta_1 P_2 D^{-1}, \end{aligned} \tag{4.22}$$

where

$$D = \rho_1\beta_1 P_2 + \rho_2\beta_2 P_4.$$

A special case of an interface is a free surface (e.g. the surface of the Earth). The coefficient for the reflection of an SH-wave from a free surface is $R_1 = 1$, as only a reflected SH-wave is generated.

For the interaction of a P or SV-wave with a free surface the reflection coefficients R_1 (P1P1), R_2 (P1S1), R_3 (S1P1) and R_4 (S1S1) are given by (Červený et al., 1977),

$$\begin{aligned} R_1 =& D^{-1}(-(1-2\beta_1^2P^2)^2 + 4P^2P_1P_2\beta_1^3\alpha_1^{-1}), \\ R_2 =& -4P\beta_1P_1(1-2\beta_1^2P^2)D^{-1}, \\ R_3 =& 4P\beta_1^2\alpha_1^{-1}P_2(1-2\beta_1^2P^2)D^{-1}, \\ R_4 =& D^{-1}(-(1-2\beta_1^2P^2)^2 + 4P^2P_1P_2\beta_1^3\alpha_1^{-1}), \end{aligned} \tag{4.23}$$

where

$$D = (1-2\beta_1^2P^2)^2 + 4P^2P_1P_2\beta_1^3\alpha_1^{-1}.$$

If the ray impinges on the free surface in the negative direction of the x-axis, the sign of the reflection coefficients R_2 and R_3 must be changed.

The interaction of a ray with an interface leads not only to a change in the direction and amplitude, but also to a discontinuity in geometrical spreading. In the case of a refracted wave (see figure 4.11) the geometrical spreading is again considered as the change in distance between two neighbouring rays. The distance between the two rays is different on the two sides of the interface. This is due to the change in orientation of the rays caused by the velocity contrast across the interface and the curvature of the interface. For the reflection the change in distance between two neighbouring rays is only due to the curvature of the interface

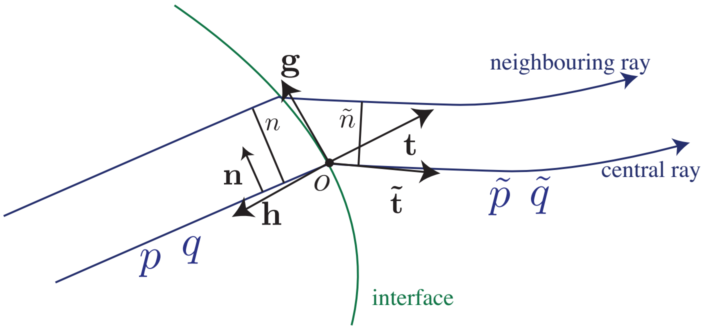


Figure 4.11: Central ray and its neighbour impinging on a curved interface. Note that the curvature of the interface and the velocity contrast lead to a change in the distance between the two refracted rays ($n \neq \tilde{n}$).

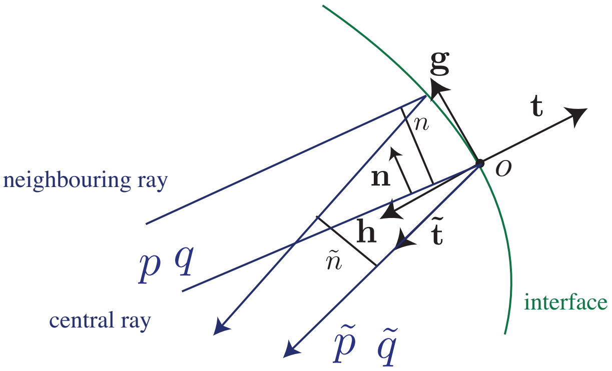


Figure 4.12: Central ray and its neighbour impinging on a curved interface. Note that the curvature of the interface leads to a change in the distance between the two reflected rays ($n \neq \tilde{n}$).

(figure 4.12).

Červený & Pšenčík (1984) provide a derivation for the equations describing the relation between $\tilde{q}$, $\tilde{p}$ and q, p at the point of intersection O:

$$\begin{aligned}\tilde{q} &= qD,\\ \tilde{p} &= [pD^{-2} + q(B + C)]D,\end{aligned} \tag{4.24}$$

where the coefficients B, C and D are given by

$$\begin{aligned}B &= [2v^{-1}\mathbf{t_g}(v^{-1}v_{\mathbf{n}}\mathbf{t_h} - \tilde{v}^{-1}\tilde{v}_{\mathbf{n}}\tilde{\mathbf{t}}_{\mathbf{h}}) - v^{-2}\mathbf{t_g}^2(v_{\mathbf{s}} - \tilde{v}_{\mathbf{s}})]\tilde{\mathbf{t}}_{\mathbf{h}}^{-2},\\ C &= G(v^{-1}\mathbf{t_h} - \tilde{v}^{-1}\tilde{\mathbf{t}}_{\mathbf{h}})\tilde{\mathbf{t}}_{\mathbf{h}}^{-2},\\ D &= \frac{\tilde{\mathbf{t}}_{\mathbf{h}}}{\mathbf{t_h}},\end{aligned}$$

with

$$\begin{aligned}\mathbf{t_g} &= \sin\nu, \quad \tilde{\mathbf{t}}_{\mathbf{g}} = \sin\tilde{\nu},\\ \mathbf{t_h} &= \cos\nu, \quad \tilde{\mathbf{t}}_{\mathbf{h}} = \cos\tilde{\nu},\end{aligned}$$

where ν and $\tilde{\nu}$ are the angles between the interface normal $\mathbf{h}$ and the ray vectors $\mathbf{t}$ and $\tilde{\mathbf{t}}$ respectively. The derivatives of the velocity v in the direction of $\mathbf{n}$ and $\mathbf{s}$ are given by

$$v_{\mathbf{n}} = v_x\mathbf{t}_y - v_y\mathbf{t}_x, \tag{4.25}$$

$$v_{\mathbf{s}} = v_x\mathbf{t}_x + v_y\mathbf{t}_y, \tag{4.26}$$

where v_x and v_y are the partial derivatives of the velocity with respect to x and y, the two axes of the Cartesian coordinate system. The terms $\mathbf{t}_x$ and $\mathbf{t}_y$ denote the x and y components of the ray vector $\mathbf{t}$, respectively.

The curvature of the interface at the point of intersection is given by G. It is positive when the interface in the vicinity of O is concave for an observer located on the incident ray. If a point (x, y) on an interface is defined as a function of path length $(x = f(s), y = f(s))$, the curvature $G(s)$ is given as

$$G(s) = \frac{y_{ss}x_s - x_{ss}y_s}{(x_s^2 + y_s^2)^{\frac{3}{2}}}, \tag{4.27}$$

where x_s and y_s denote the first derivatives of the two position variables with respect to s and x_{ss} and y_{ss} the second derivatives. These derivatives are given by analytical expressions when the interfaces are described using a cubic B-spline approximation (appendix B).

4.3 Examples

Two examples are presented to illustrate the efficiency and robustness of the wavefront tracking scheme in media which contain interfaces. In the first example the wavefront tracker is used to compute travel times and paths of global phases. In the second example it is shown how the combination of wavefront tracking with the Gaussian beam method can be used to yield information about the structure beneath a receiver. The algorithms have been implemented under GNU/Linux in Fortran and all computation times are given for a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory.

4.3.1 Global travel time model

The ak135 model (Kennett et al., 1995) is a one dimensional reference model of the earth. Accurate tables of arrival times (`http://rses.anu.edu.au/seismology/ttsoft.html`) are compared with solutions calculated using the wavefront tracking approach presented in this work. The computations are performed on a spherical earth. This implies that no corrections for ellipticity are applied to either the reference values for the travel time or the solutions computed using the wavefront tracking scheme. The model parameterization that is used allows for the description of complex models as illustrated earlier (figure 4.5) in a Cartesian framework. To mimic a spherical medium, the basic idea is to sample the ak135 model on a Cartesian grid and describe the layered earth using isolated bodies and overturning interfaces. Provided a high enough grid resolution for the underlying Cartesian grid is used, the error associated with the conversion of the model from polar to Cartesian coordinates should be negligible.

For an intermediate depth event (300 km) the travel times and ray paths for different global phases are computed. The travel times are compared with the reference travel times for the ak135 model. The core mantle boundary and the inner-outer core boundary are represented by explicit interfaces, and the grid spacing for the underlying velocity field is 5 km. Mantle discontinuities and the Moho are represented by sharp velocity gradients rather than explicit interfaces. The

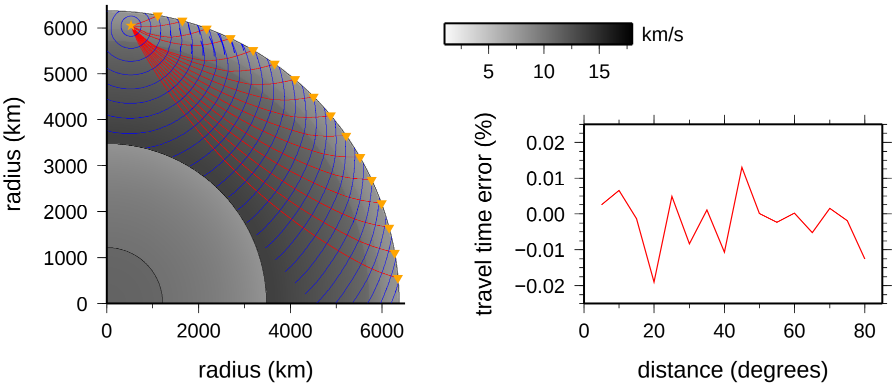


Figure 4.13: Ray paths, wavefronts and relative travel time error for the direct P-phase in the ak135 global model. Computation time is 231 s.

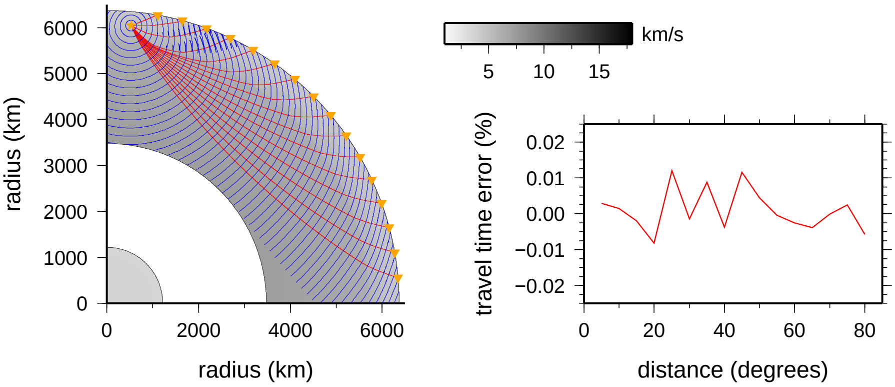


Figure 4.14: Ray paths, wavefronts and relative travel time error for the direct S-phase in the ak135 global model. Computation time is 243 s.

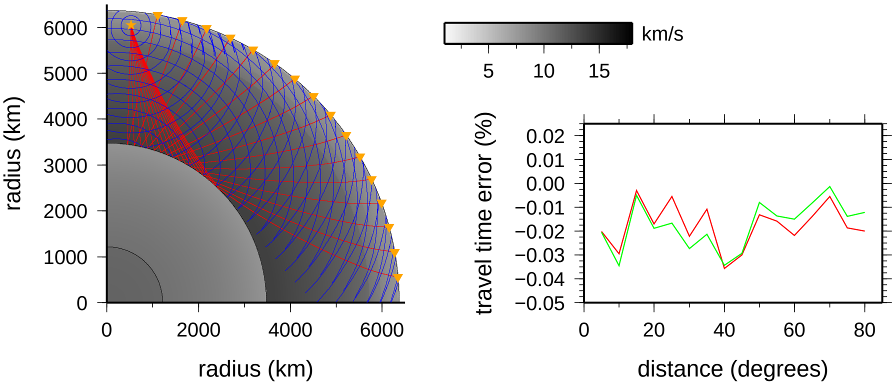


Figure 4.15: Ray paths, wavefronts and relative travel time error for the PcP-phase in the ak135 global model. Computation time is 242 s. Two curves are given for the travel time error in this figure. The green curve is for an interface grid comprising 2000 × 2000 nodes and the red curve for an interface grid comprising 1000 × 1000 nodes

time step is set to 0.2 s and 100 points are used to represent the bicharacteristic strip at the source. As outlined in section 4.1.1 the interfaces are sampled on to an interface grid as part of the wavefront tracking scheme. In this example the interface grid is given by 1000 × 1000 nodes.

The ray paths, wavefronts and relative travel time error for the direct P and S-phases are shown in figures 4.13 and 4.14. The errors are comparable to those of the constant velocity gradient case shown earlier (section 3.2.2). Note that for the S-wave, the wavefronts triplicate due to the velocity contrast between crust and mantle.

The travel times for the PcP and PcS-phases (figures 4.15 and 4.16), which are reflections from the core mantle boundary, show a slightly larger travel time error than the direct waves. The results computed using the wavefront tracker appear to underestimate the travel times. Increasing the resolution of the interface grid helps to reduce the error for rays with larger epicentral distances. However one should keep in mind that applying cubic B-spline approximation to a model which has been converted into a Cartesian coordinate system can lead to a different velocity field compared to applying linear interpolation (which is consistent with the official definition of ak135) to the original model formulated in polar coordinates, where the velocity is a function of the radius. Another source of error is that in the model used for the wavefront tracker mantle discontinuities and the Moho are not represented by explicit interfaces.

Looking at the wavefronts computed in figure 4.16d, it can be observed that the scheme does not to predict PcS-phases for epicentral distances of 60 ° or larger, while the ak135 model predicts PcS-phases for angular distances of up to 65 °. If a point on the wavefront hits the interface (in this case the core-mantle boundary) and its neighbour does not hit the interface, no search for the point between the two nodes, which would still hit the interface, is performed. For the chosen point density PcS-phases for angular distances larger than $\sim 57.5\,^\circ$ cannot be computed. Increasing the point density represents a short term solution. Ultimately, a search for endpoints would need to be added to the wavefront tracking scheme to alleviate this problem.

It is also possible to compute phases which have interacted with more than one interface. The wavefronts and ray paths for a $PKiKP$-phase, a P-wave which has bounced from the inner-outer core boundary are given in figure 4.17. Again the results computed using the wavefront tracking approach seem to underestimate the travel times. This is likely due to the same reasons discussed for the PcP and PcS

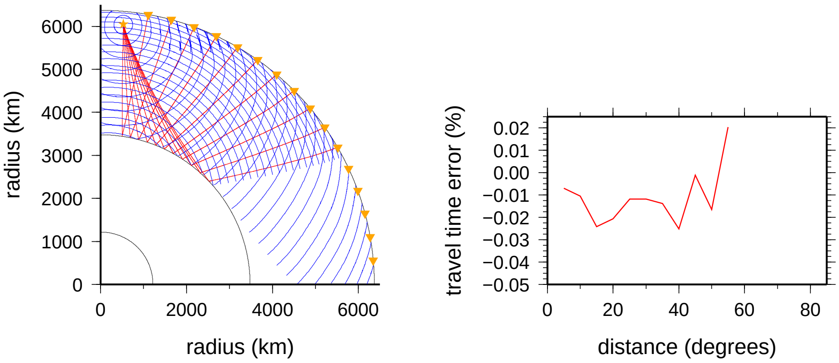


Figure 4.16: Ray paths, wavefronts and relative travel time error for the PcS-phase in the ak135 global model. Computation time is 198 s.

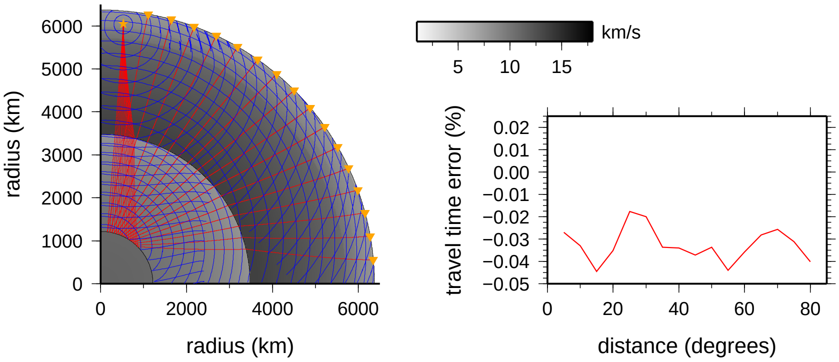


Figure 4.17: Ray paths, wavefronts and relative travel time error for the $PKiKP$-phase in the ak135 global model. Computation time is 263 s.

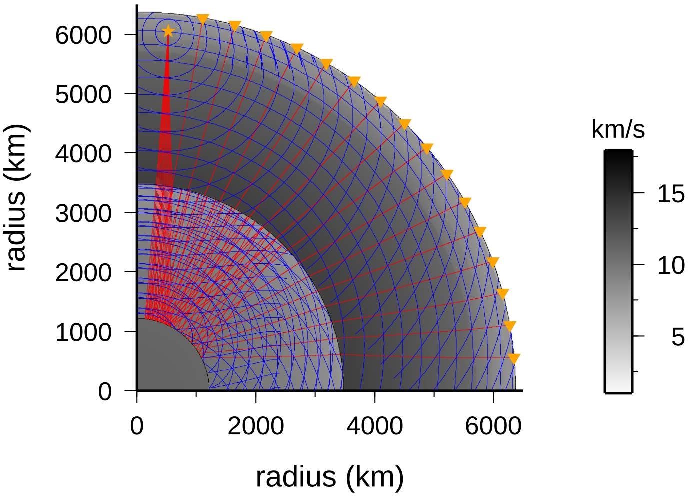


Figure 4.18: Ray paths and wavefronts for the $PKiKKiKP$-phase in the ak135 global model. Computation time is 289 s.

phases. Note that the wavefront tracking scheme can also compute travel times for more complex phases like *PKiKKiKP* (figure 4.18). However, if a higher accuracy is desired for a spherical model the algorithm should be modified to work natively in spherical coordinates.

In figure 4.19 Gaussian distributed random numbers with a standard deviation of 2.5 km/s are created on a grid with a spacing of 400 km. They are then added to the ak135 Cartesian velocity grid using cubic B-spline approximation. The position of the core mantle boundary is also perturbed in the depth direction using Gaussian random numbers with a standard deviation of 100 km. *P* and *PcP*-phases are computed for this model. The structure gives rise to several occurrences of multipathing (see figure 4.19) due to the shape of the core mantle boundary and velocity heterogeneities in the mantle.

Computation times for all global phases calculated here are of the order of minutes and no longer just seconds. This is due to the ray path extraction which is based on the wavefronts being stored on disk and no longer kept in memory as it is the case for the scheme used in chapter 3. The computation of the *PcS*-phase takes less time than the calculation of the *PcP*-phase, because the number of receivers registering a *PcS*-wave is smaller. If only arrival times are required for the *PcP*-phase the computation time is only 44 s. The computation time for the example where Gaussian distributed random noise has been added to the structure is significantly larger, because the wavefronts are more complex and several later arrivals are generated, for which a ray path has to be extracted. If however, only wavefronts and arrival times are desired (i.e. no ray paths are computed) the computation time is only 77 s, compared to 1053 s when ray paths have to be calculated.

This example demonstrates that wavefront tracking is not limited to computing travel times for local or regional models; it can also be used to compute travel times for global phases. The accuracy is of the order of 0.1 s which is comparable to the accuracy achieved for global seismic phases when a graph and perturbation method are used (e.g. Bijwaard & Spakman, 1999). Compared to conventional ray methods that account for lateral heterogeneity in the earth (e.g. Julian & Gubbins, 1977; Thurber & Ellsworth, 1980) this new approach has the advantage of robustness (as shown in figure 4.19), and potentially, efficiency .

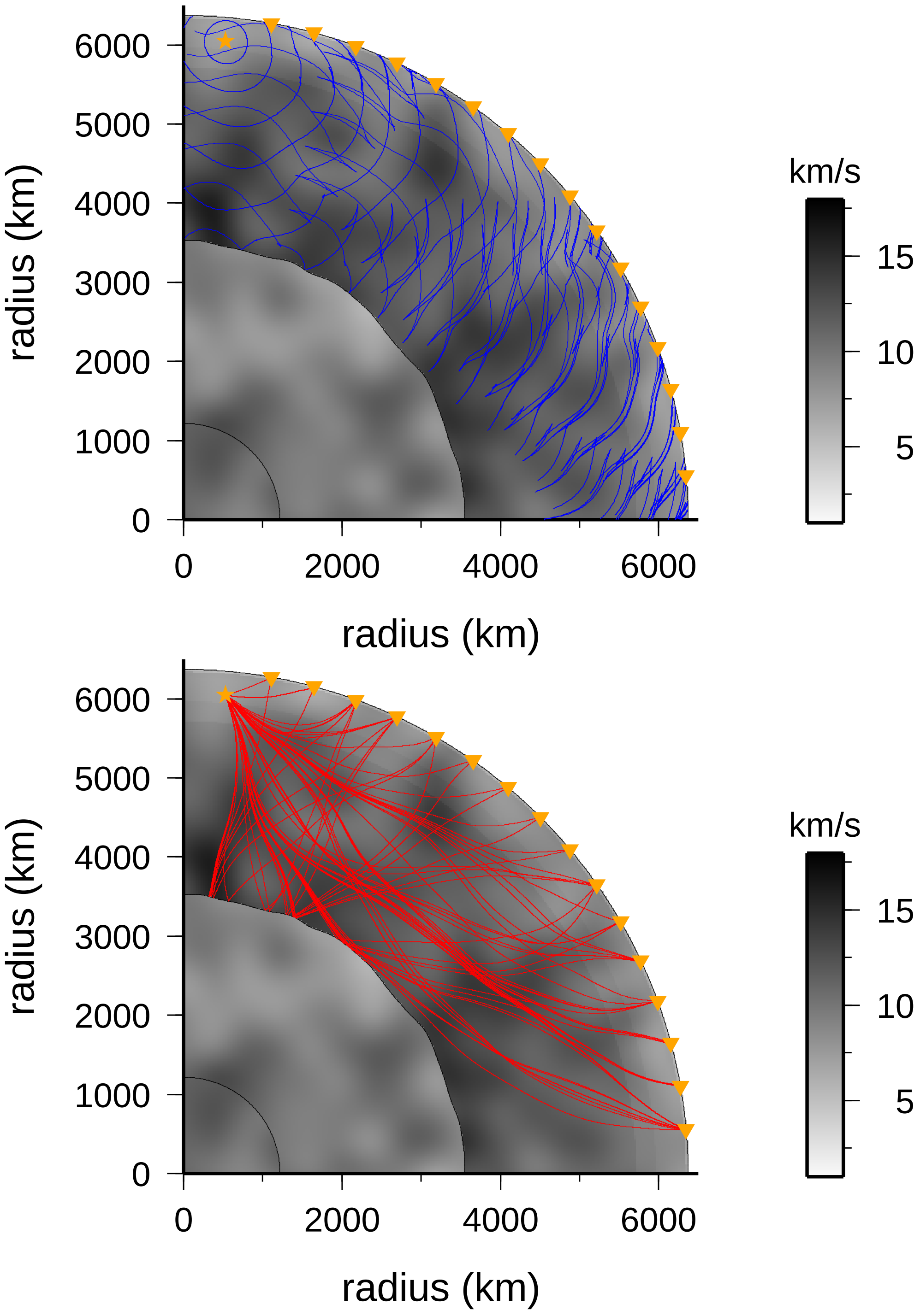


Figure 4.19: Wavefronts (top) and ray paths (bottom) for P and PcP-phases computed for the ak135 model with Gaussian distributed random noise added to the velocity field and core mantle boundary geometry. Computation time is 1053 s when ray paths are extracted. If only arrival times are computed the computation time reduces to 77 s.

	P-wave speed (km/s)	density (g/cm^3)	thickness (km)
layer 1	5.75	2.61	15
layer 2	6.75	2.93	15
layer 3	8.20	3.39	-

Table 4.1: The velocity, density and thickness of the three layers in the model used to compare the Gaussian beam method with the reflection matrix approach.

4.3.2 Receiver functions

In receiver function analysis, the response of structure beneath a receiver to an incoming plane wave is used to image the subsurface (e.g. Ammon, 1991). Estimating receiver functions typically involves the computation of synthetic seismograms. These seismograms can, for example, be computed using the reflection matrix approach of Kennett (1983) (e.g. Tomlinson et al., 2006). For more complex structures kinematic and dynamic ray tracing (e.g. Du & Foulger, 1999) or even finite difference solutions of the wave equation (e.g. Hammer & Langston, 1996) can be used to obtain amplitude and travel time information.

Another approach involves using the Gaussian beam method for the computation of synthetic seismograms generated by an incoming plane wave. When compared to standard ray tracing or the reflectivity method, this has the advantage that later arrivals of the different phases are automatically included.

In a first example, the impulse response obtained by an application of the Gaussian beam method is compared to the reflectivity method (Kennett, 1983) for a structure with three horizontal layers and an incoming P-wave propagating in the vertical direction. The initial beam width for the Gaussian beams is $0.025\,\mathrm{km}^{1/2}$. The velocity values, densities and thickness of the three layers are given in table 4.1. No P to S conversions are generated in this example due to the use of a normal incidence P-wave. The maximum number of interface interactions a phase can have is limited to 8, which means that in total 15 phases are tracked.

Figure 4.20 shows the two impulse responses; one computed using the reflection matrix approach and the other computed using the Gaussian beam method. The first thing to note is that the direction of the displacement of the two sets of phases is the same. Overall the same eight arrivals in both impulse responses can be identified. The amplitude varies slightly between the two schemes, but considering that the results of the Gaussian beam method are to some degree influenced by the choices of the parameters, the correlation between the two time

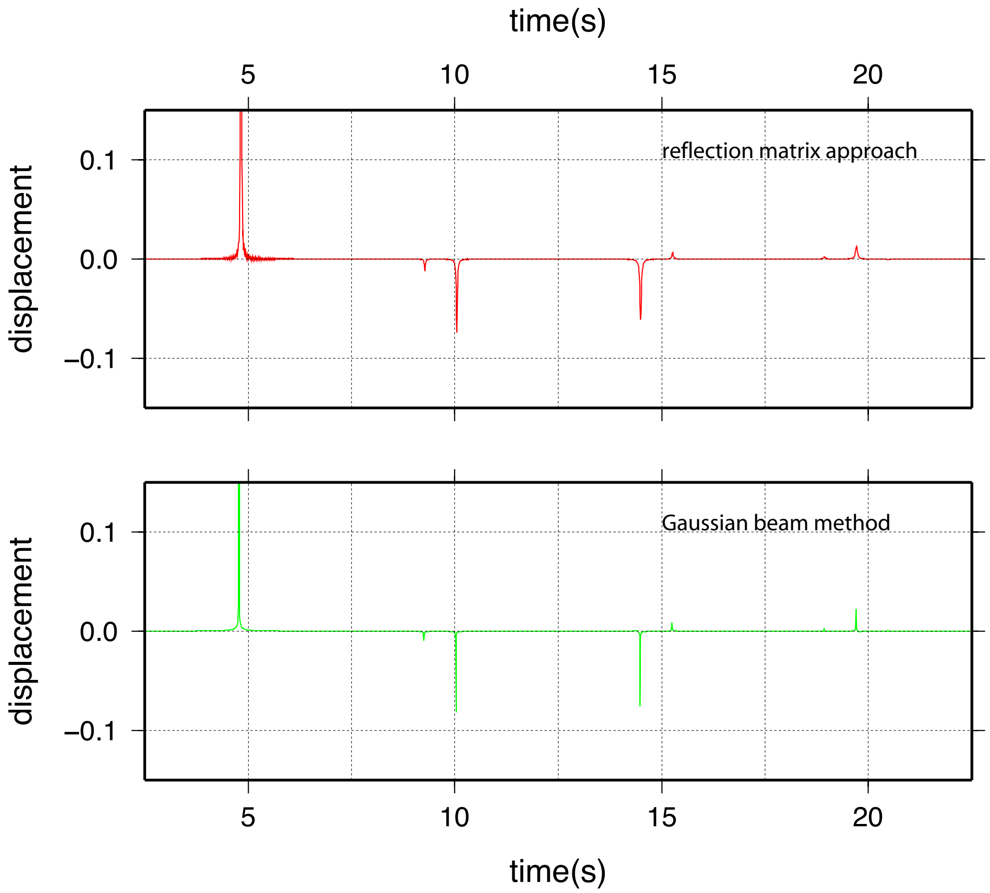


Figure 4.20: Impulse response for an incoming plane wave in a model with three horizontal layers (table 4.1). The time series have been normalised so that the maximum absolute displacement is one.

series is good. The sample rate for both signals is 100 Hz. The pulses appear to be sharper when the Gaussian beam method is used and there is less noise on the signal, especially near the first arrival. The Gaussian beam method can therefore be seen as a viable alternative for generating synthetic seismograms for receiver function analysis. The advantage of the approach proposed here, when compared to the reflection matrix approach, is that it accounts for lateral variations in structure and includes multi arrival information if it is present.

Figure 4.21 shows a structure with two layers and the ray paths associated with an incoming plane wave. The P and S-wave velocity increases linearly with depth in the top layer and is constant in the bottom layer. The P-wave speed at the surface is 4.5 km/s and the velocity gradient is 0.025 s^{-1}. The S-wave speed at the surface is 3.5 km/s and the velocity gradient is 0.01 s^{-1}. In the bottom layer

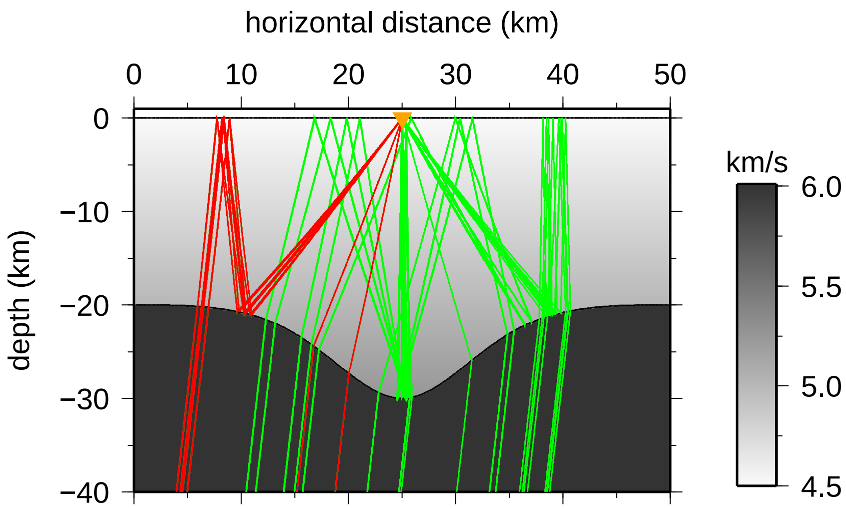


Figure 4.21: Ray paths for an incoming plane wave and its reverberations in the upper layer. The P-wave speed is plotted. The first arrival ray paths for the different phases are plotted in red, and the ray paths for the later arrivals in green.

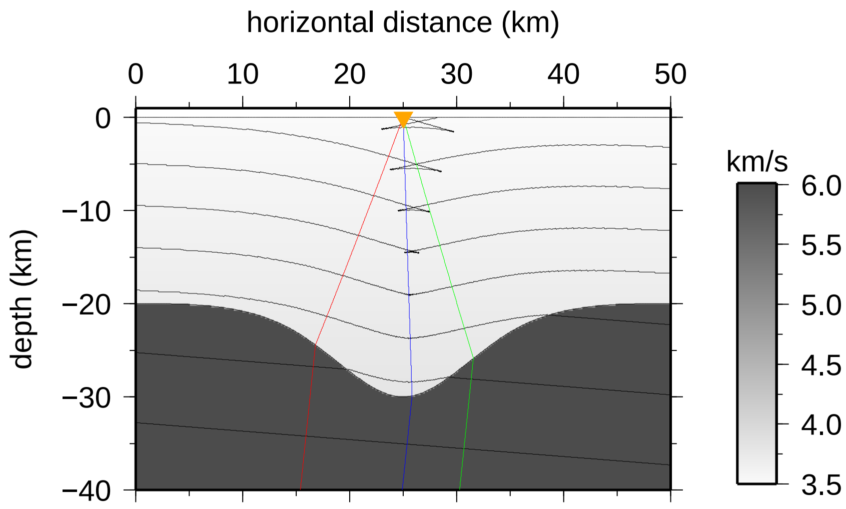


Figure 4.22: Ray paths and wavefronts for the P to S conversion of the incoming plane wave. The wavefront triplicates due to the shape of the interface and the velocity contrast. The velocity values shown in the figure are the P-wave speed in the lower layer and the S-wave speed in the upper layer. The ray path for the first arrival is red, for the second arrival green, and for the third arrival blue.

the P-wave velocity is 6.0 km/s and the S-wave velocity is 4.4 km/s. The density in the upper layer is 2.55 g/cm^3 and in the lower layer 2.93 g/cm^3.

An incoming teleseismic P-wave is represented by a plane wave with 200 nodes on the bicharacteristic strip and 250×250 nodes are used for the interface grid. A time step of 0.025 s is used. The incoming P-wave is split at the interface into a P and S-wave. In addition to the direct P-wave and the P to S conversion, all reverberations of the incoming P-wave which bounce once between the surface and the interface are computed. This means that in total 10 different phases are simulated. Due to the velocity structure the wavefronts triplicate and in addition to the 10 first arrivals of the different phases 30 later arrivals are also generated. Figure 4.22 illustrates, for the P to S conversion of the incoming plane wave, how later arrivals are generated due to the shape of the interface and the velocity contrast. The wavefront triplicates for this phase and three arrivals are generated.

The Gaussian beam method is used to compute synthetic seismograms with an initial beam width of 0.005 km$^{1/2}$ and a peak angular frequency for the Ricker wavelet of $\omega_{peak} = 0.5\,\mathrm{rad}\cdot\mathrm{s}^{-1}$. The computation time for the wavefront tracking and the calculation of a synthetic seismogram is 115 s. Figure 4.23 shows the waveform if only first arrivals of the different phases are used, and figure 4.24 if first and later arrivals are used during the computation of the waveform. As the direct P-wave does not triplicate, the wavepackages associated with the first arrivals in the two seismograms have the same shape. The shape of the P to S conversion is, however, significantly different depending on whether first arrivals or first and later arrivals are used. This is due to the triplication of the P to S conversion as discussed previously.

For the two waveforms, it is possible to measure their sensitivity to changes in structure. The bottom of the valley in the interface is defined by the depth of the control node at a horizontal distance of 25 km. The depth of this node is chosen as the independent parameter which will be used to investigate the sensitivity of the waveform. Note that due to the cubic B-spline representation of the interfaces, the control node does not necessarily lie on the interface (see figure 4.21). First, the waveform for the reference depth of 35 km is computed, and then the depth is varied and the difference between the seismogram for the current depth and the reference depth is calculated. The difference between two waveforms is measured by computing the RMS value for the difference between the two seismograms, after having normalised them so that the displacement amplitudes lie in $[-1, 1]$. In figure 4.25 the horizontal components of the synthetic seismograms

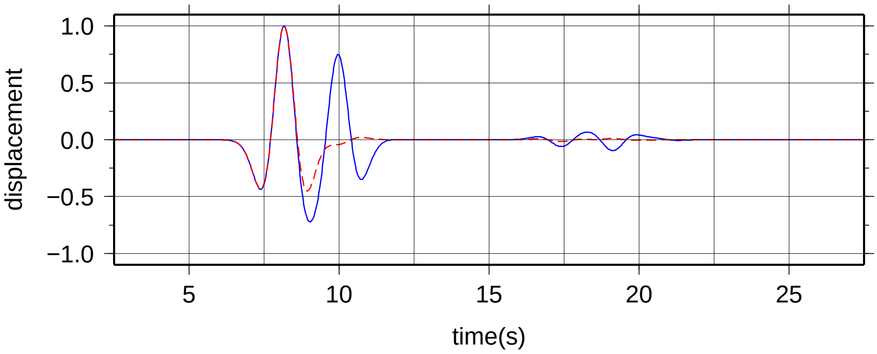


Figure 4.23: Synthetic seismograms based on first arrivals of the different phases. The vertical (dashed red line) and horizontal component (blue line) have been normalised separately so that the maximum displacement is one.

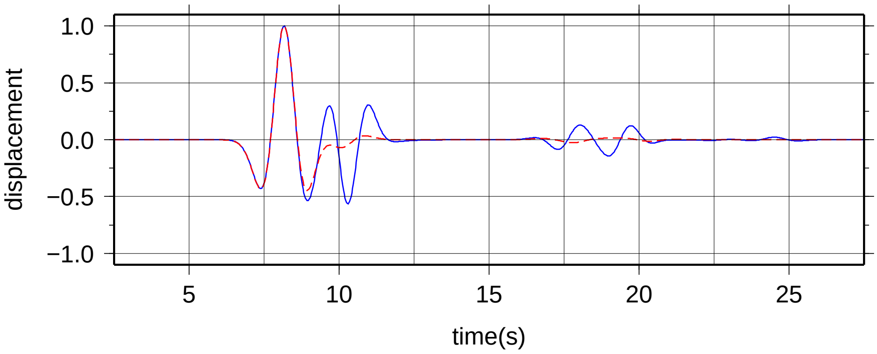


Figure 4.24: Synthetic seismograms based on first and later arrivals of the different phases. The vertical (dashed red line) and horizontal component (blue line) have been normalised separately so that the maximum displacement is one.

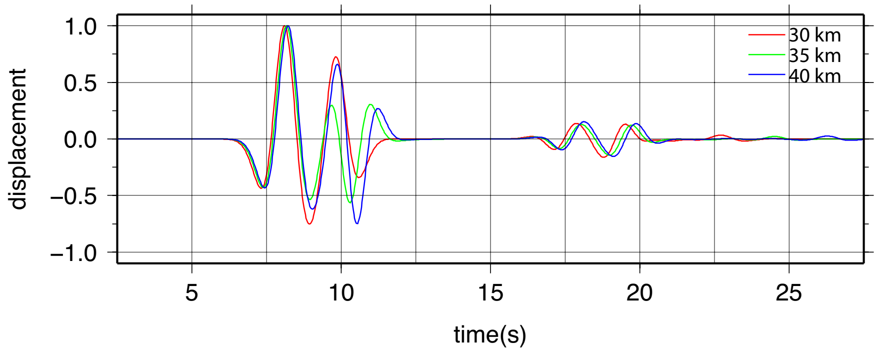


Figure 4.25: Horizontal components of the synthetic seismograms for different depths of the central interface control node, when first and later arrivals are used in the Gaussian beam method. Each waveform has been normalised separately so that the maximum displacement is one.

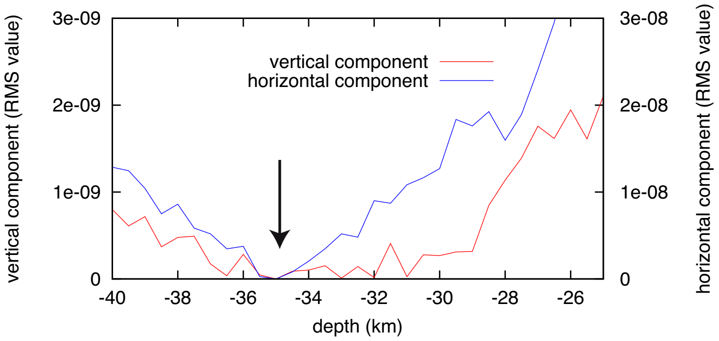


Figure 4.26: RMS value of the difference between the two components of the seismograms, when only first arrivals are used. In this and the following two figures the global minimum (reference depth of 35 km) is marked by a black arrow.

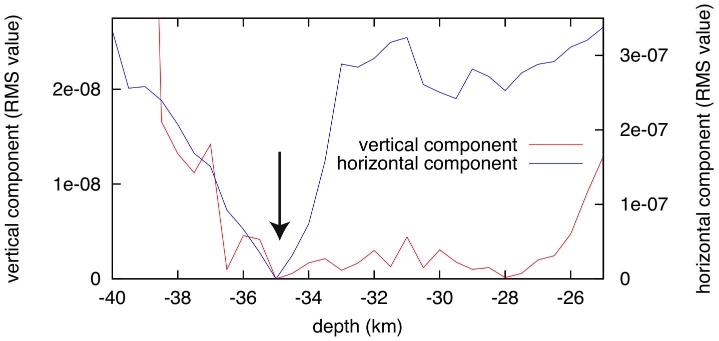


Figure 4.27: RMS value of the difference between the two components of the seismograms, when first and later arrivals are used. Note that a different scaling is used when compared to figure 4.26.

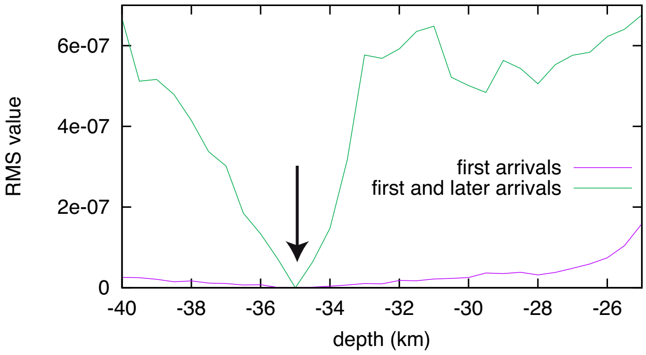


Figure 4.28: Combined RMS values of the vertical and horizontal components for the two scenarios. The violet curve denotes the difference between seismograms based on first arrivals, and the green curve denotes the difference between seismograms based on first and later arrivals.

are plotted for three different depths of the central interface control node when first and later arrivals are used. Clearly the shape of the first arrival wavepackage changes significantly when the depth of the central interface node is varied.

Figure 4.26 shows the RMS value for the vertical and horizontal component if only first arrivals are used. For a depth of 35 km (i.e. the depth of the reference model) the RMS value is 0 for both components. The RMS values when first and later arrivals are used in the computation of the synthetic seismograms is given in figure 4.27. In this case the global minimum (black arrow) is better defined compared to when only first arrivals are used. This is also shown in figure 4.28 where the RMS values for the vertical and horizontal component have been combined for the two approaches. Clearly the RMS value and hence the waveform is much more sensitive to small changes in structure when first and later arrivals are simulated. Hence the later arrivals contain additional structural information.

4.4 Summary

The Lagrangian solver has been modified to cope with the explicit presence of interfaces in a velocity medium. In addition, the Gaussian beam method has been implemented for the computation of ray based synthetic seismograms. The scheme can now be used to propagate wavefronts in two dimensional structures which may contain pinch-outs, isolated bodies and overturning interfaces. Reflected and refracted phases and their later arrivals can successfully be tracked. It is, however, not possible at this stage to track critical refracted waves, which travel along an interface.

It has been shown that the wavefront tracking approach can be used to successfully compute travel times for global phases, which have interacted with the core mantle and inner-outer core boundary, and also later arrivals associated with these phases. The scheme can also handle non spherical boundaries, e.g. topography of the core mantle boundary (e.g. Morelli & Dziewonski, 1987).

The Gaussian beam method (e.g. Nowack & Aki, 1984), normally requires a wide fan of rays to be shot. Only a small portion of these rays actually contribute to the wavefield at the receiver. Using wavefront tracking to determine source-receiver paths means that it is possible to shoot narrower fans of rays and therefore avoid the computation of rays which do not contribute to the displacement field at the receiver.

The receiver function example demonstrates that there is tangible later arrival information contained in the waveforms, as computed using Gaussian beams. The scheme could for example be used in a region with significant interface topography. Langston & Hammer (2001) discuss vertical receiver functions for stations in the Los Angeles Basin and Long Valley Caldera containing secondary arrivals with large amplitudes, which cannot be explained by a simple one dimensional velocity structure, but are an indication of multiple ray focusing in a three dimensional basin structure. Multi valued wavefront tracking could help to understand the distortion of the wavefield especially once the method is extended for three dimensional structures.

In the next chapter, numerical experiments will be carried out in order to demonstrate how travel time information contained in later arrivals can be used in seismic tomography.

Chapter 5

Seismic tomography with later arrivals

The inversion of first arrival travel times is undoubtedly the most popular technique for imaging subsurface seismic structure at all scales (e.g. Nolet, 1987; Bregman et al., 1989; Toomey et al., 1998; Rawlinson et al., 2006b; Conder & Wiens, 2006). A comprehensive review of the methodology and its application has been given by Rawlinson & Sambridge (2003).

In seismic tomography, one usually starts with an initial set of model parameters $\mathbf{m}_0$ (e.g. the wave speed at each node of a grid, and the depths of a set of interface control nodes). The aim is then to adjust the model parameters so that the predicted travel times $\mathbf{g}(\mathbf{m})$ fit the observed travel times $\mathbf{d}_{obs}$. The functional $\mathbf{g}$, however, is non-linear because the ray path depends on the velocity structure. One approach to mitigating this problem involves using a linear approximation, based on the derivatives of $\mathbf{g}$ with respect to the model parameters, to iteratively reduce the difference between the observed and predicted travel times $\delta\mathbf{d} = \mathbf{g}(\mathbf{m}) - \mathbf{d}_{obs}$. The forward problem for the change in the travel time residuals $\delta\mathbf{d}$ as a function of the change in the model parameters $\delta\mathbf{m}$ can then be written as

$$\delta\mathbf{d} = \mathbf{G}\delta\mathbf{m}. \tag{5.1}$$

The Fréchet matrix $\mathbf{G}$ is an $n \times m$ matrix, where n is the number of rays and m the number of model parameters; G_{ij} is thus given by the travel time derivative of the i-th ray with respect to the j-th model parameter. The linear system of equations (5.1) is rarely even determined (i.e. a square system of $n \times n$ or $m \times m$ ($n = m$) linearly independent equations), so direct solution for $\delta\mathbf{m}$ is usually not possible.

One common approach to overcoming this problem is to compute the model perturbation $\delta\mathbf{m}$ using singular value decomposition or SVD (Aster et al., 2005). Singular value decomposition allows the matrix $\mathbf{G}$ to be factored as follows (e.g. Press et al., 1992):

$$\mathbf{G} = \mathbf{U}\mathbf{S}\mathbf{V}^T, \tag{5.2}$$

where $\mathbf{U}$ is an $m \times m$ orthogonal matrix whose columns are the basis vectors spanning the data space R^m, $\mathbf{V}$ an $n \times n$ orthogonal matrix with columns that are the basis vectors spanning the model space R^n and $\mathbf{S}$ an $m \times n$ diagonal matrix whose positive diagonal elements are the singular values. The singular values along the diagonal of $\mathbf{S}$ are generally arranged in descending order. It can be shown that every matrix has a singular value decomposition. SVD can be used to compute a generalised inverse of the Fréchet matrix $\mathbf{G}$, called the Moore-Penrose pseudoinverse (Aster et al., 2005), given by

$$\mathbf{G}^{\dagger} = \mathbf{V}\mathbf{S}^{-1}\mathbf{U}^T. \tag{5.3}$$

The so called pseudoinverse solution for $\delta\mathbf{m}$ is then defined as

$$\begin{aligned}\delta\mathbf{m} &= \mathbf{G}^{\dagger}\delta\mathbf{d},\\ &= \mathbf{V}_p\mathbf{S}_p^{-1}\mathbf{U}_p^T\delta\mathbf{d},\\ &= \sum_{i=1}^{p}\frac{(\mathbf{U}_{.,i})^T\delta\mathbf{d}}{s_i}\mathbf{V}_{.,i},\end{aligned} \tag{5.4}$$

where $\mathbf{U}_{.,i}$ and $\mathbf{V}_{.,i}$ denotes the vector given by the i-th column of the matrices $\mathbf{U}$ and $\mathbf{V}$ respectively. The subscript p denotes that only the first p singular values of $\mathbf{S}$ are nonzero and therefore only the first p columns of $\mathbf{U}$ and $\mathbf{V}$ are used. The remaining columns can be ignored as they are multiplied with zero singular values. This is shown in the last line of (5.4), where the summation is carried out over the first p columns of $\mathbf{U}$ and $\mathbf{V}$. It is common to decrease p and eliminate model space vectors associated with small singular values. The solution then becomes more stable. However this stability comes at the expense of reducing the subspace of R^n in which the solution lies. As a result, the fit to the data may worsen. It can be shown that using the generalised inverse provides a solution for $\delta\mathbf{m}$ that always exists (Aster et al., 2005). For an over determined system of equations the least squares solution for $\delta\mathbf{m}$ is found if the pseudoinverse is used.

Once the change in the model parameters $\delta\mathbf{m}$ has been computed, the model can be updated and a new set of predicted travel times computed. This process can be repeated until the travel time residuals fall below a given tolerance, or the solution converges. The problem with (5.4) is that it requires the singular value decomposition of the Fréchet matrix $\mathbf{G}$, which tends to be a very large sparse matrix for most real applications.

In practice, a priori information such as data and initial model uncertainty is often included. A reality of all seismic imaging problems is that they are under or mix determined, which means that more than one solution will satisfy the observations. Solving equation (5.1) using the pseudo inverse has the potential to yield just one solution from a wide spectrum of candidate models. Therefore, additional constraints are usually necessary to reduce the pool of potential solutions to those that exhibit certain desirable characteristics. This process, referred to as regularisation, can be achieved by formulating the problem as one of objective function minimisation.

5.1 Objective function

The objective function in seismic tomography consists of a data residual term and one or more regularisation terms. A commonly used form of the objective function $S(\mathbf{m})$ is (Rawlinson & Sambridge, 2003)

$$S(\mathbf{m}) = \frac{1}{2}[\Psi(\mathbf{m}) + \epsilon\Phi(\mathbf{m}) + \eta\Omega(\mathbf{m})], \tag{5.5}$$

where $\Psi(\mathbf{m})$ is a measure of the difference between observed and predicted travel times, $\Phi(\mathbf{m})$ a model damping term, ϵ the damping factor, $\Omega(\mathbf{m})$ a model smoothing term and η the smoothing factor.

The term $\Psi(\mathbf{m})$ defines the difference between the observed and predicted data in the least squares or L_2 sense:

$$\Psi(\mathbf{m}) = (\mathbf{g}(\mathbf{m}) - \mathbf{d}_{obs})^T \mathbf{C}_d^{-1} (\mathbf{g}(\mathbf{m}) - \mathbf{d}_{obs}), \tag{5.6}$$

where $\mathbf{C}_d$ is the data covariance matrix, $\mathbf{g}(\mathbf{m})$ the model prediction and $\mathbf{d}_{obs}$ the observed data. If the errors are assumed to be uncorrelated, the data covariance matrix is diagonal - $\mathbf{C}_d = [\delta_{ij}(\sigma_j)^2]$ - and its diagonal elements are given by the uncertainty in the observed travel time σ_j associated with the j-th ray. Thus, in this context, $\mathbf{C}_d$ is essentially a data weighting matrix.

If the problem is mixed-determined or under-determined, not all model parameters will be well constrained by the data alone. A damping term $\Phi(\mathbf{m})$ in the objective function provides additional constraints on the model parameters by penalising solution models that depart significantly from the reference model $\mathbf{m}_0$ and therefore reduces the non-uniqueness of the solution. The damping term $\Phi(\mathbf{m})$ is typically defined as

$$\Phi(\mathbf{m}) = (\mathbf{m} - \mathbf{m}_0)^T \mathbf{C}_m^{-1} (\mathbf{m} - \mathbf{m}_0), \tag{5.7}$$

where $\mathbf{C}_m$ is an a priori model covariance matrix. If the errors are assumed to be uncorrelated $\mathbf{C}_m$ is a diagonal matrix and its diagonal elements correspond to the uncertainty associated with each model parameter. One can therefore refer to $\mathbf{C}_m$ as the model weighting matrix.

The minimum structure solution attempts to find a compromise between satisfying the data and finding a model with a minimum amount of structural variation (Constable et al., 1987). This requirement can be included in the objective function by using the term (e.g. Sambridge, 1990)

$$\Omega(\mathbf{m}) = \mathbf{m}^T \mathbf{D}^T \mathbf{D} \mathbf{m}, \tag{5.8}$$

where $\mathbf{Dm}$ is a finite difference estimate of specified spatial derivatives between model parameters.

Minimisation of (5.5) will produce a solution that satisfies the data but is to some degree smooth and not strongly perturbed from the initial model $\mathbf{m}_0$. The parameters ϵ and η, which control model perturbation and roughness, are usually chosen on an ad hoc basis. While this is not ideal, the underlying philosophy of trying to produce a smooth, minimally perturbed solution at least reduces the prospect of over-interpretation by the end-user.

Gradient based inversion methods use the derivative of the objective function $S(\mathbf{m})$ at a specific point in model space as a basis for locating an improved model. The basic assumption is that $S(\mathbf{m})$ is sufficiently smooth to allow a local quadratic approximation in the neighbourhood of the current model:

$$S(\mathbf{m} + \delta\mathbf{m}) \approx S(\mathbf{m}) + \boldsymbol{\gamma}\delta\mathbf{m} + \frac{1}{2}\mathbf{m}^T \mathbf{H} \delta\mathbf{m}, \tag{5.9}$$

where $\delta\mathbf{m}$ is a perturbation to the current model, $\boldsymbol{\gamma} = \partial S / \partial \mathbf{m}$ is the gradient vector and $\mathbf{H} = \partial^2 S/\partial \mathbf{m}^2$ the Hessian matrix, the derivative of the gradient vector. For the objective function (5.5), the gradient vector $\boldsymbol{\gamma}$ is given by

$$\boldsymbol{\gamma} = \mathbf{G}^T \mathbf{C}_d^{-1}[\mathbf{g}(\mathbf{m}) - \mathbf{d}_{obs}] + \epsilon \mathbf{C}_m^{-1}(\mathbf{m} - \mathbf{m}_0) + \eta \mathbf{D}^T \mathbf{D}\mathbf{m}, \tag{5.10}$$

and the Hessian matrix $\mathbf{H}$ is defined as

$$\mathbf{H} = \mathbf{G}^T \mathbf{C}_d^{-1}\mathbf{G} + \nabla_{\mathbf{m}} \mathbf{G}^T \mathbf{C}_d^{-1}[\mathbf{g}(\mathbf{m}) - \mathbf{d}_{obs}] + \epsilon \mathbf{C}_m^{-1} + \eta \mathbf{D}^T \mathbf{D}. \tag{5.11}$$

The second derivative term in the expression for the Hessian Matrix can be neglected since it is time consuming to evaluate and its effect is small if the residuals $\mathbf{g}(\mathbf{m}) - \mathbf{d}_{obs}$ are already small or if the forward problem is quasi linear ($\nabla_{\mathbf{m}} \mathbf{G} \approx 0$).

The aim is then to find a model perturbation $\delta \mathbf{m}_i$ which minimises the objective function (5.5). Since $\mathbf{g}(\mathbf{m})$ is generally non-linear, the minimisation of the objective function $S(\mathbf{m})$ requires an iterative approach,

$$\mathbf{m}_{i+1} = \mathbf{m}_i + \delta \mathbf{m}_i. \tag{5.12}$$

5.2 Subspace method

The model perturbation $\delta \mathbf{m}_i$ (see equation (5.12)) is computed in this thesis using the subspace method. One of the advantages of the subspace method is that the pseudoinverse has to be computed for only a small matrix. It is also often claimed that the subspace method tends to converge faster than steepest descent and conjugate gradient due to its simultaneous search along several directions that together span a subspace of the model space (e.g. Kennett et al., 1988; Williamson, 1990; Rawlinson & Sambridge, 2003).

In the subspace method, the minimisation of the quadratic approximation of $S(\mathbf{m})$ is restricted to a p dimensional subspace of model space. The perturbation $\delta \mathbf{m}$ then occurs in the space spanned by a set of p M dimensional basis vectors $\mathbf{a}^j$

$$\delta \mathbf{m} = \sum_{j=1}^{p} \boldsymbol{\mu}_j \mathbf{a}^j = \mathbf{A}\boldsymbol{\mu}, \tag{5.13}$$

where $\mathbf{A} = [\mathbf{a}^j]$ is the $m \times p$ projection matrix, and m is the number of model parameters. The components $\boldsymbol{\mu}_j$ of the vector $\boldsymbol{\mu}$ determine the length of the corresponding vectors $\mathbf{a}^j$ in $\mathbf{A}$ that minimises the quadratic approximation to $S(\mathbf{m})$

(5.9) in the subspace spanned by the vectors $\mathbf{a}^j$. If the p dimensional subspace has the same dimensions as the model space, $\boldsymbol{\mu}$ and $\mathbf{A}$ are given by the pseudoinverse solution (5.4):

$$\begin{aligned}\mathbf{A}\boldsymbol{\mu} &= \sum_{i=1}^{p} \frac{(\mathbf{U}_{.,i})^T \delta\mathbf{d}}{s_i} \mathbf{V}_{.,i} \\ &= \mathbf{V}_p \mathbf{S}_p^{-1} (\mathbf{U}_p)^T \delta\mathbf{d} \\ &= \mathbf{G}^{\dagger} \end{aligned} \tag{5.14}$$

However, the aim is to avoid having to compute a pseudoinverse of the large matrix $\mathbf{G}^{\dagger}$. In practice, the dimension of the subspace is chosen to be significantly smaller than the dimension of the model space. Substituting (5.13) into (5.9) leads in summation form to

$$S(\mathbf{m} + \delta\mathbf{m}) \approx S(\mathbf{m}) + \sum_{j=1}^{p} \mu_j \boldsymbol{\gamma}^T \mathbf{a}^j + \frac{1}{2} \sum_{j=1}^{p} \sum_{k=1}^{p} \mu_j \mu_k [\mathbf{a}^k]^T \mathbf{H} [\mathbf{a}^j]. \tag{5.15}$$

The minimum of $S(\mathbf{m})$ with respect to $\boldsymbol{\mu}$ is then given as

$$\frac{\partial S(\mathbf{m})}{\partial \mu_q} = \boldsymbol{\gamma}^T \mathbf{a}^q + \sum_{k=1}^{p} \mu_k [\mathbf{a}^k]^T \mathbf{H} [\mathbf{a}^j] = 0, \tag{5.16}$$

for $q = 1, .., p$. Rearranging (5.16) for $\boldsymbol{\mu}$ gives

$$\boldsymbol{\mu} = -[\mathbf{A}^T \mathbf{H} \mathbf{A}]^{-1} \mathbf{A}^T \boldsymbol{\gamma}. \tag{5.17}$$

Since $\delta\mathbf{m} = \mathbf{A}\boldsymbol{\mu}$ and $\mathbf{H} \approx \mathbf{G}^T \mathbf{C}_d^{-1} \mathbf{G} + \epsilon \mathbf{C}_m^{-1} + \eta \mathbf{D}^T \mathbf{D}$, the model perturbation $\delta\mathbf{m}$ is given by

$$\delta\mathbf{m} = -\mathbf{A}[\mathbf{A}^T (\mathbf{G}^T \mathbf{C}_d^{-1} \mathbf{G} + \epsilon \mathbf{C}_m^{-1} + \eta \mathbf{D}^T \mathbf{D}) \mathbf{A}]^{-1} \mathbf{A}^T \boldsymbol{\gamma}. \tag{5.18}$$

This equation can be used iteratively in the manner specified by (5.12) to reduce the travel time residuals. The projection matrix $\mathbf{A}$, the Fréchet derivatives $\mathbf{G}$, the model travel times $\mathbf{g}(\mathbf{m})$, and therefore the gradient vector $\boldsymbol{\gamma}$, are re-evaluated between successive iterations. The basis vectors (i.e. the columns of the projection matrix $\mathbf{A}$) are often constructed using the steepest ascent vector in model space $\mathbf{C}_m \boldsymbol{\gamma}$ and its rate of change (e.g. Kennett et al., 1988; Sambridge, 1990; Williamson, 1990).

The subspace method has the desirable property that for a p dimensional subspace only a $p \times p$ matrix has to be inverted, where p is usually small. For example, Williamson (1990) uses a six dimensional subspace, which means that the inverse of a 6×6 matrix is computed. In the implementation used in this work, the pseudoinverse of this matrix is computed using singular value decomposition (Aster et al., 2005), since it can have a large condition number i.e. zero or very small eigenvalues (Sambridge, 1990).

If one is dealing with multiple parameter classes, such as velocity and interface position, the subspace method offers a natural way to deal with their simultaneous inversion. The basis vectors $\mathbf{a}^j$ can be chosen such that each vector lies in the space spanned by a particular parameter class. The minimisation of $S(\mathbf{m})$ will then account for the different sensitivities of the objective function with respect to the different parameter classes in a balanced way (Kennett et al., 1988). Given a model with an interface and velocity field, one can choose a set of two basis vectors, corresponding to the two different parameter classes, that are to be constrained by the data. The two orthogonal search directions can simply be obtained by partitioning the gradient vector in model space $\mathbf{C}_m\boldsymbol{\gamma}$ on the basis of each parameter class as

$$\mathbf{C}_m\boldsymbol{\gamma} = \mathbf{a}^1 + \mathbf{a}^2 = \begin{bmatrix} \boldsymbol{\gamma}^1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \boldsymbol{\gamma}^2 \end{bmatrix}, \tag{5.19}$$

where each of the two ascent vectors $\mathbf{a}^1$ and $\mathbf{a}^2$ lie in the space of only one parameter type. Additional basis vectors can be obtained by pre-multiplying $\mathbf{a}^1$ and $\mathbf{a}^2$ with the model space Hessian $\mathbf{C}_m\mathbf{H}$. This can also be done when only one parameter class is present. By repeating the pre-multiplication of the latest set of basis vectors with the model space Hessian, additional basis vectors can be constructed. Increasing the dimension of the subspace will improve convergence, but this comes at a greater computational cost per iteration.

Once the desired set of basis vectors (i.e. columns of $\mathbf{A}$) is obtained, orthogonalisation is applied in order to avoid interdependence between the basis vectors. Singular value decomposition of the projection matrix $\mathbf{A}$ is used to find the set of orthonormal basis vectors. If the singular value for one or more of the resulting basis vectors is close to zero, the subspace spanned by the set of basis vectors is in fact less than p dimensions. The subspace dimension p should then be reduced so that the basis vectors have singular values different to zero by at least a small tolerance. In this implementation of the subspace method, at each inverse step one starts with an upper limit for p, which is then reduced until all the singular values of the projection matrix **A** are larger than a predetermined threshold (i.e. the basis vectors spanning the subspace are linearly independent of each other).

Due to the sparseness of the matrices $\mathbf{G}$, $\mathbf{A}$, $\mathbf{C}_m$, $\mathbf{C}_d$ and $\mathbf{H}$, compact row storage is used when implementing the subspace method. When a matrix is described using compact row storage, only the non zero elements, their column index and an index of the values which begin a new row, are stored. This significantly reduces the memory requirements. All matrix operations required in the subspace method, except the inversion of the $p \times p$ matrix, are implemented for matrices in compact row storage format. As well as reducing memory, computation time is also reduced as the calculations are only done using the non zero elements of the matrices. However since the forward step is much more time consuming (i.e. tracking the wavefront, extracting the rays and computing the Fréchet derivatives), the time spent in solving the inverse step is in any event marginal.

5.3 Fréchet matrix

Once the ray path for a given arrival is known, the change of the arrival time with respect to a change in a model parameter can be computed. The linearised relationship between travel time residual and velocity perturbation is given by (e.g. Rawlinson & Sambridge, 2003)

$$\delta t = -\int_{L(v)} \frac{\delta v}{v^2}\,\mathrm{d}l, \tag{5.20}$$

where δv is the velocity perturbation, $L(v)$ the ray path and v the reference velocity model. If the velocity field is defined by a grid of nodes, then to first order, the Fréchet derivatives can be written as

$$\frac{\partial t}{\partial v_n} = -\int_{L(v)} v^{-2}\frac{\partial v}{\partial v_n}\,\mathrm{d}l, \tag{5.21}$$

where v_n is the velocity of a particular node and $\partial v/\partial v_n$ is the change of velocity along the ray segment $\mathrm{d}l$ with respect to a change in v_n. If the velocity interpolation or approximation function has a simple form (e.g. cubic B-spline approximation), this expression is straightforward to calculate (see appendix D.1).

First order accurate analytic expressions can also be obtained for the Fréchet derivatives of travel times with respect to interface grid points. The basic approach is to partition the problem by applying the chain rule of differentiation, and then separately calculating each element. The derivative of the travel time with respect to a change in the depth coordinate z_{int} of an interface control point can be written as

$$\frac{\partial t}{\partial z_n} = \frac{\partial t}{\partial h_{int}} \frac{\partial h_{int}}{\partial z_{int}} \frac{\partial z_{int}}{\partial z_n}, \tag{5.22}$$

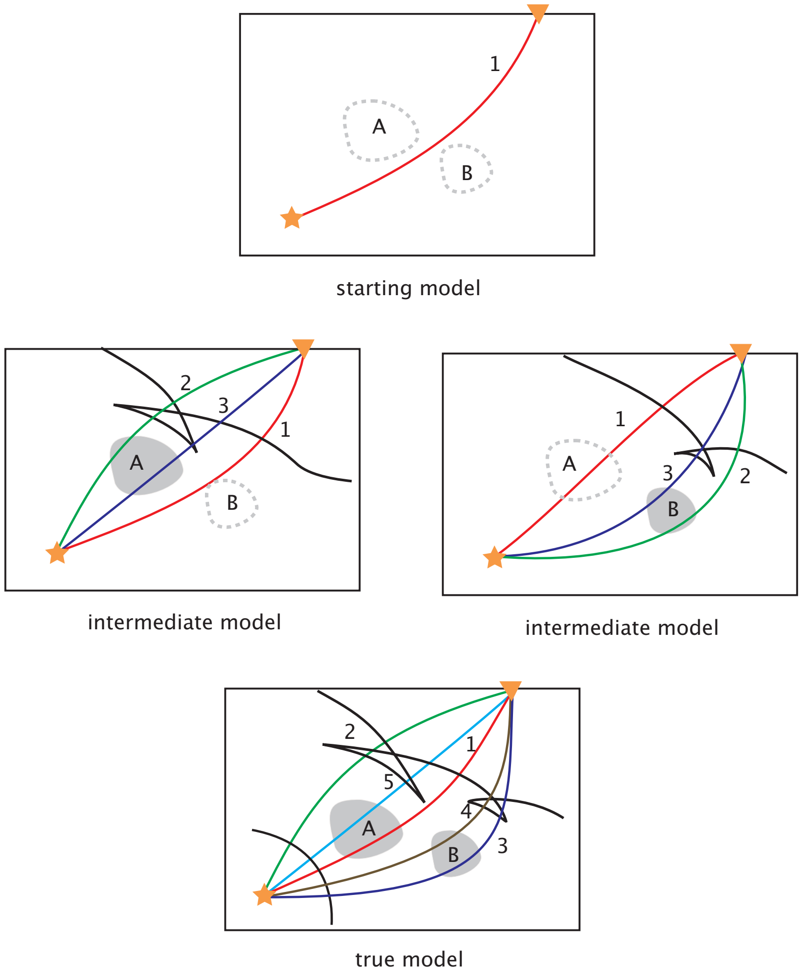
where z_n is the depth coordinate of the interface node and h_{int} is the displacement in the direction of the interface normal at the intersection point (see appendix D.2). By replacing z_{int} with x_{int} one can write an expression for the derivative of the travel time with respect to a change in the horizontal coordinate of an interface control point. Having the capability to move the interface control points in both directions is important if overturning interfaces are to be considered.

5.4 Including later arrivals in tomography

In chapters 3 and 4, it was claimed that later arrivals contain additional information about the structure through which they propagate. This means that if they are used in seismic tomography, improved images should result. To investigate this possibility, several numerical tests are performed. In the first test, the aim is to recover a smooth velocity model (similar to those in chapter 3) using first arrivals only, and then both first and later arrivals. In the second test, velocity and interface structure will be recovered simultaneously.

It is important at this stage to emphasise that the existence of later arrivals is to a much greater extent a function of the model than the existence of first arrivals. In a smooth velocity model, there is always a first arrival ray path between a source and a receiver, but later arrivals only exist if velocity gradients cause the wavefront to form a swallowtail. Here, later arrivals refers to the multipathing of transmissions and reflections, and not just the first arrivals of reflections and refractions at discontinuities, which are sometimes used in tomography and also referred to as later arrivals (e.g. Rawlinson & Sambridge, 2003).

In seismic tomography one typically begins with a simple, often one dimensional, starting model (e.g. Graeber et al., 2002; Conder & Wiens, 2006; Rawlinson et al., 2006b). These models are unlikely to generate the multipathing observed in real data because of their simplicity. However, as the iterative inversion progresses and the model more closely resembles the true structure, these previously redundant later arrivals will gradually be activated and used to refine the solution. This means that during the iterative inversion procedure, the number of ray



starting model

intermediate model

intermediate model

true model

Figure 5.1: Scenario where five arrivals are observed but only three are predicted by an intermediate model during an iterative inversion procedure. It is not clear which three of the five later arrivals actually correspond to the predicted arrivals.

paths and hence data is not constant. A consequence of this phenomenon is that the RMS (root mean square) data prediction error can actually increase as later arrivals appear for the first time and, at least initially, are not well described by the model.

Once later arrivals start to be predicted during an iterative inversion procedure they have to be correctly matched with observations. For example, given a particular source-receiver combination with five observed and three predicted arrivals, it may not be clear which two of the four observed later arrivals actually correspond to the predicted later arrivals. This is illustrated in more detail in figure 5.1 for one source-receiver pair. The structure is dominated by two low velocity anomalies A and B, which cause the development of two swallowtails. Five arrivals are observed at the receiver. Let us now assume that in the first step of an iterative inversion procedure either anomaly A or B is recovered. This means that the intermediate model predicts only three arrivals, while five arrivals are observed. It is not clear which three of the five observed later arrivals actually corresponds to the predicted arrivals. They are therefore matched according to their number, which means that the first predicted later arrival is assigned to the first observed later arrival and so on until there are no predictions or observations left. Although this may result in incorrect phase associations in some cases, the underlying improvement of the model at each iteration should gradually reduce this possibility. In the early stages of an iterative inversion procedure one would also expect the first arrivals to be the dominant influence in the reconstruction of model perturbations. Therefore a few mismatched later arrivals should not have a significant influence. As the solution converges towards the true model, additional later arrivals may appear, but they are more likely to be matched correctly.

The forward step of the iterative inversion procedure is solved using the Lagrangian scheme discussed in chapters 3 and 4. The inverse step has been implemented under GNU/Linux in Fortran using compact row storage for the matrices. The forward and inverse step are separate applications. An iterative inversion procedure can then be set up using a scripting language to call the two applications in sequence until the RMS error of the data misfit is below a certain threshold. The velocity model, the travel times and the Fréchet derivatives are passed between the forward and inverse step by writing them on disk, with the Fréchet derivatives stored in a binary file.

5.4.1 Smooth velocity model

A smooth velocity model is used in order to verify whether or not the inclusion of later arrivals can improve seismic imaging. A low velocity anomaly is recovered using two approaches: (1) the inversion only uses first arrivals, (2) the inversion uses first and later arrivals. The true model (i.e. the model we will try to recover) and ray paths used in the tests are shown in figure 5.2. The paths are generated by 15 earthquake sources at depth, and terminate at an array of 13 receivers at the surface. Note how most of the first arrivals avoid the low velocity region, unlike the later arrivals, especially those that arrive last. For the given source-receiver configuration, 195 first arrivals, 35 second arrivals and 35 third arrivals are generated. These arrivals form the set of observed travel times for the synthetic model

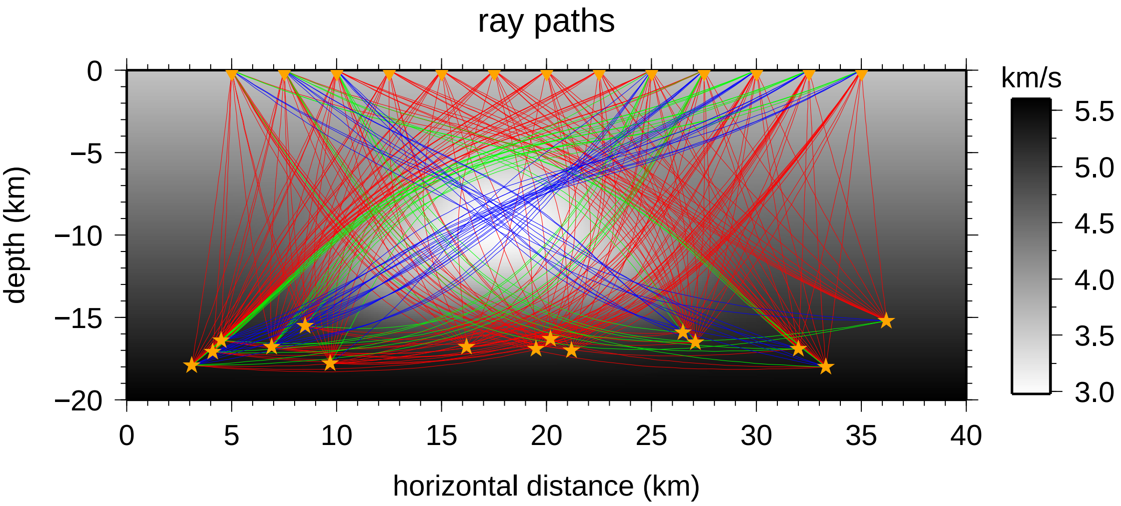


Figure 5.2: Synthetic velocity model and ray paths used to test the ability of later arrivals to improve tomographic imaging. The ray paths corresponding to the first, second and third arrivals are plotted as red, green and blue lines, respectively. Stars denote sources and triangles denote receivers.

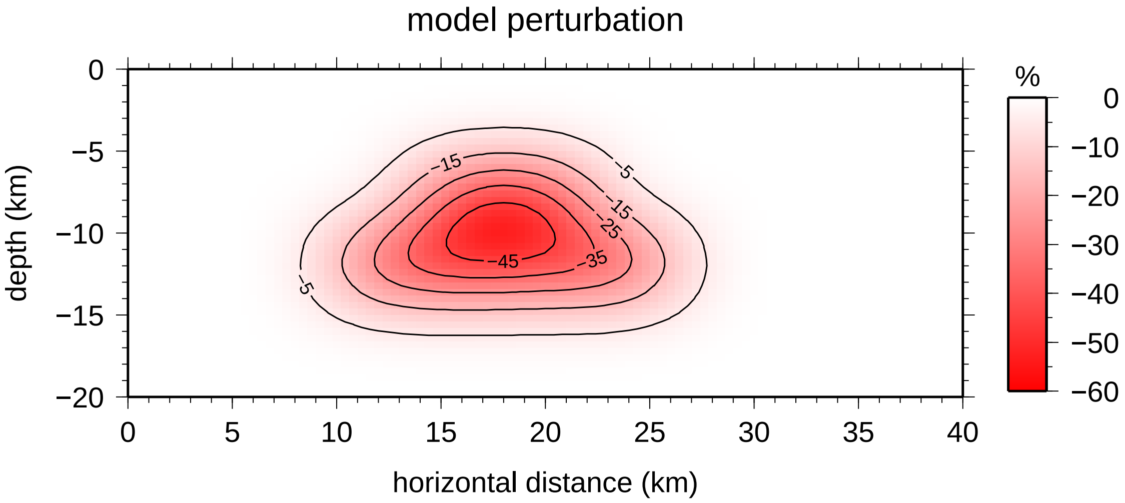


Figure 5.3: Velocity anomaly plotted as a model perturbation with respect to the starting model (cf. figure 5.2).

the recovery of which will be attempted using the iterative inversion procedure. The number of second and third arrivals is equal because in such a simple model, only one swallowtail is formed, which means that if a second arrival is observed at a receiver, a third arrival must also be observed.

The model, which is described by 11×6 nodes, has a velocity of 3.6 km/s at the surface and 5.6 km/s at the base with a constant vertical gradient. A low velocity anomaly is then superimposed onto this background model. The perturbation of the low velocity anomaly with respect to the background velocity model is shown in figure 5.3. The peak amplitude of the anomaly is 60 %, which is very large. The limited vertical and horizontal extent of the area covered by the model, and the smooth large scale velocity variations, means that a large amplitude is needed to cause multipathing between several sources and receivers. One should also keep in mind that the region where the amplitude is above 45 % is small compared to the size of the model. The background one dimensional velocity distribution is used as the initial model in the iterative inversion procedure.

For both the first arrival only and the first and later arrival test cases, the inversion is performed using a four dimensional subspace, which is reduced when the basis vectors are not orthonormal. The iterative inversion procedure is stopped once the RMS error for the travel time residuals is below 0.02 s. Note that no regularisation is used in these tests.

Figures 5.4 and 5.5 show the number of later arrivals and the RMS difference between the observed and predicted travel times as a function of iteration. The solution model computed using only first arrivals generates 18 later arrivals. When later arrivals are used in the inversion, the RMS error no longer monotonically decreases, which is due to the gradual appearance of later arrivals that are yet to be used to constrain structure (figure 5.4). This is illustrated in figure 5.6, where the RMS error for the different arrivals is shown as a function of iteration. The RMS error for the first arrivals rapidly decreases in the first two iterations and then begins to undulate. For the second and third arrivals it initially undulates and only eventually decreases. Although an additional three iterations are required to satisfy the stopping criterion when later arrivals are used, the final number of predicted later arrivals of 62 is much closer to the 70 that are observed. It is interesting to note that the number of later arrivals does not necessarily increase from one iteration to the next, and their introduction or removal tends to be gradual rather than dramatic.

Figures 5.7 and 5.8 show the relative error between the inversion result and true

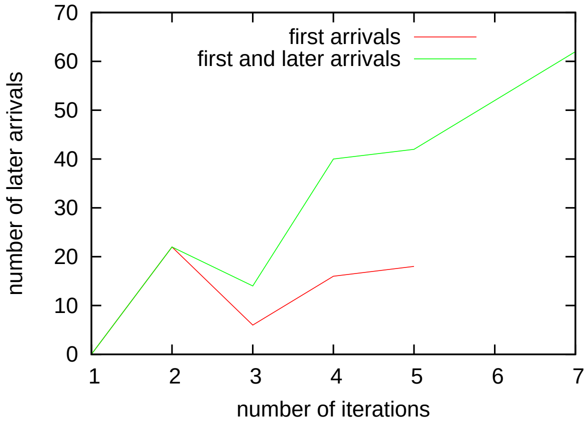


Figure 5.4: The number of predicted later arrivals as a function of iteration for the two solutions.

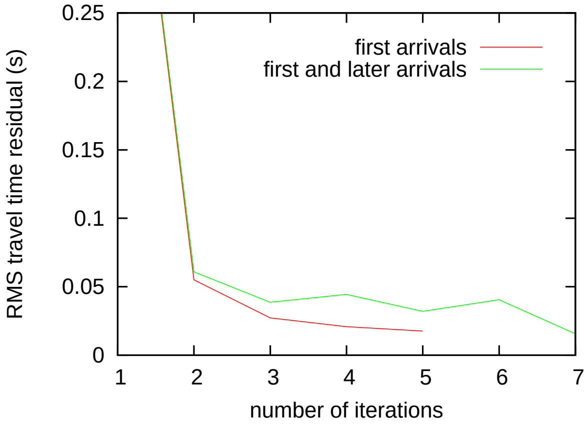


Figure 5.5: RMS travel time residuals for the two solutions.

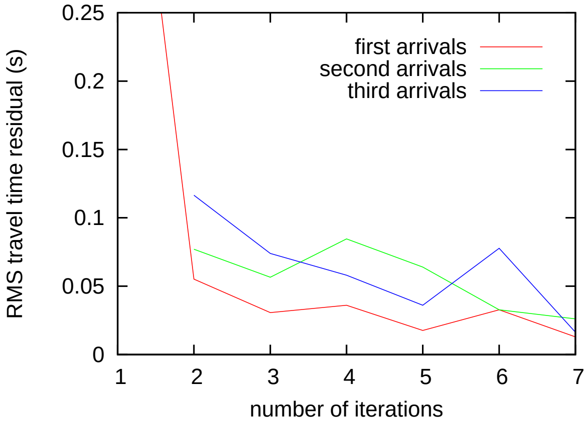


Figure 5.6: RMS travel time residuals for the different arrivals when first and later arrivals are used in the iterative inversion procedure.

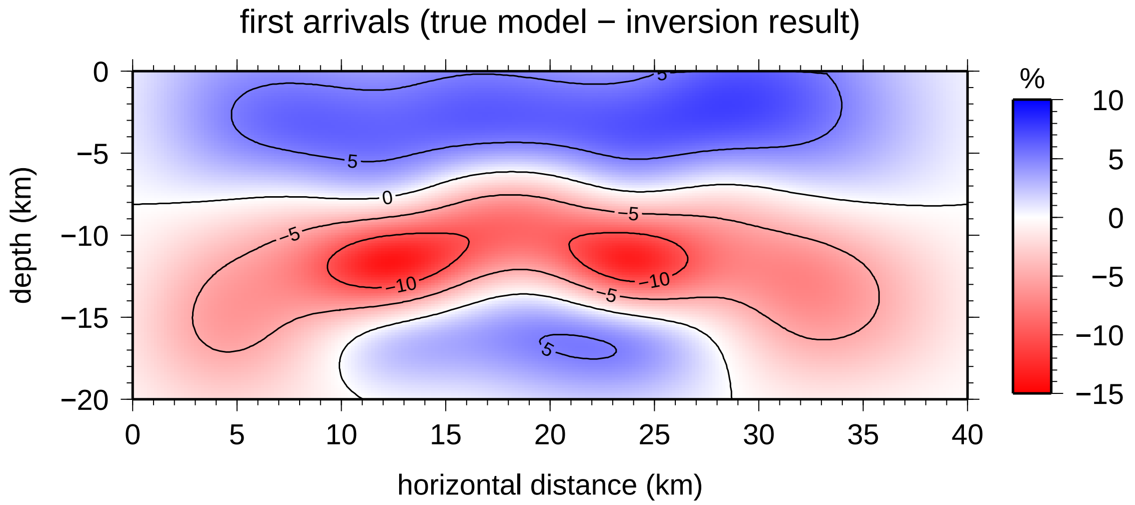


Figure 5.7: Relative error in percent between the inversion result based on first arrivals and the true model.

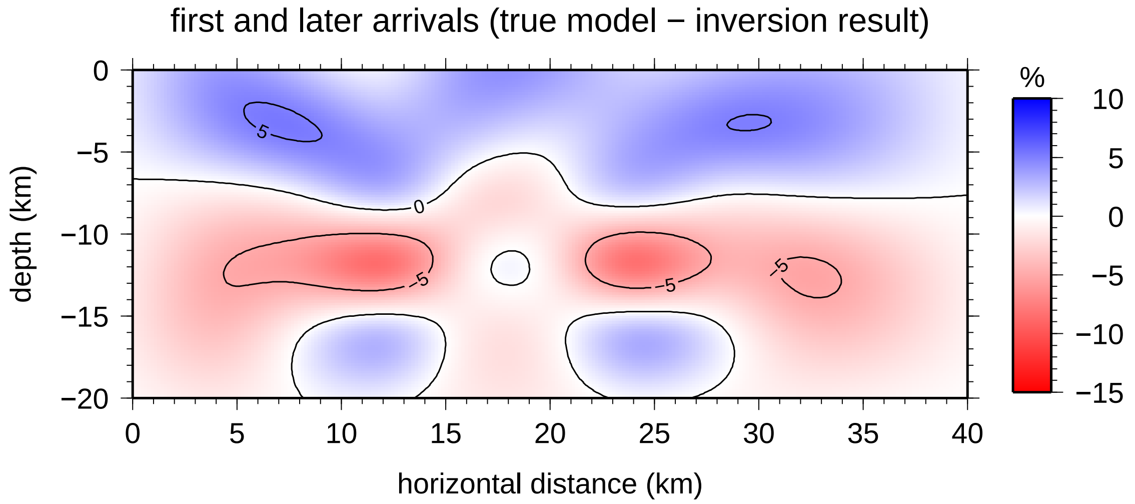


Figure 5.8: Relative error in percent between the inversion result based on first and later arrivals and the true model.

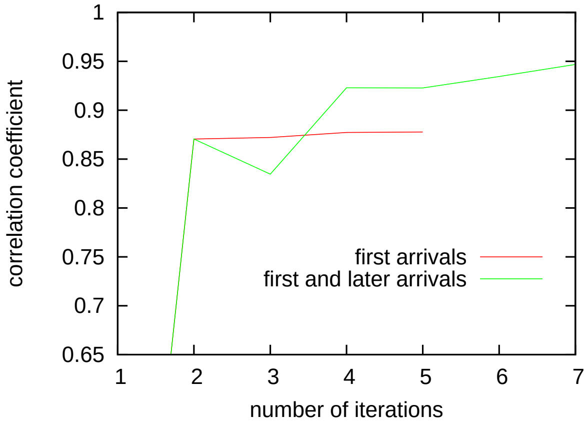


Figure 5.9: Correlation coefficient between the true anomaly and the solution anomalies for the two solutions.

model in percent for each of the two cases. If first and later arrivals are used the overall error is smaller and the centre of the anomaly, where the ray paths for the third arrivals cross each other, is better resolved. The quality of the inversion result can be quantified by calculating the correlation coefficient between the true and inverted velocity anomalies. If only first arrivals are used the correlation coefficient is 0.88. The inclusion of later arrivals increases the model correlation coefficient to 0.95. Clearly, the recovery of the low velocity anomaly is greatly improved when both first and later arrivals are exploited in the inversion. Figure 5.9 shows the correlation coefficient between the true model and the inversion result as a function of iteration. The later arrivals initially decrease the correlation coefficient and only in later iterations do they actually increase.

For a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory and GNU/Linux as the operating system, the computation time for the iterative inversion procedure when first arrivals are used is 2 min and 34 s, which compares to 4 min and 32 s when first and later arrivals are used. The multi arrival tomography takes significantly longer because of the larger number of iterations, and also because of the larger number of arrivals for which ray paths and Fréchet derivatives have to be computed. However, in terms of the correlation coefficient (figure 5.9), four iterations of the multi arrival tomography procedure are sufficient to obtain an inversion result superior to that achieved by first arrival tomography in five iteration steps.

The above example illustrates that using later arrivals in seismic tomography can help to improve the quality of the recovered images. On the other hand, the appearance and disappearance of later arrivals can cause some undulation of the RMS error during the iterative process, which may need to be prolonged in order to achieve convergence. In the next example, seismic imaging of a layered velocity model using later arrivals is investigated, where it is shown in much more detail how later arrivals influence the objective function.

5.4.2 Layered velocity model

In this numerical test the inversion is performed simultaneously for the velocity and interface structure of a two layered model using: (1) only first arrivals and (2) first and later arrivals. The true model (i.e. the model we try to recover) and the ray paths are shown in figure 5.10. Note how the later arrivals are clustered in the valley structure created by the downward deflection of the interface. For the two sources in the top layer, the direct wave and the reflection from the interface are used, and for the two sources in the bottom layer, the refracted wave and the associated multiple (i.e. reflected once between free surface and interface). Two incoming plane waves which refract at the interface are also included. The number of observed arrivals is provided in table 5.1. This forms our set of observed travel times for the synthetic model, which we will try to recover in an iterative inversion procedure.

first arrivals	170	fourth arrivals	12
second arrivals	32	fifth arrivals	12
third arrivals	32	sixth arrivals	2
total (1st, 2nd and 3rd)	234	total (4th, 5th and 6th)	26

Table 5.1: Number of observed arrivals for the test problem.

The synthetic test model is based on a depth dependent background velocity field. In the upper layer the velocity at the surface is 3.0 km/s and increases with depth at a rate of $0.05\,\mathrm{s}^{-1}$. In the lower layer, the velocity increases at a rate of $0.01\,\mathrm{s}^{-1}$ to 8.2 km/s at the bottom. The interface is given by a horizontal line. The spacing of the velocity grids is 10 km and the interface is defined using 9 control nodes. A perturbation to both interface and velocity structure is then superimposed on the background model as shown in figure 5.11. There are two low velocity anomalies (15% perturbation) in the upper layer and the central interface node is perturbed so that a valley is formed. The background velocity distribution is used as the initial model for the inversion procedure. A 12 dimensional subspace

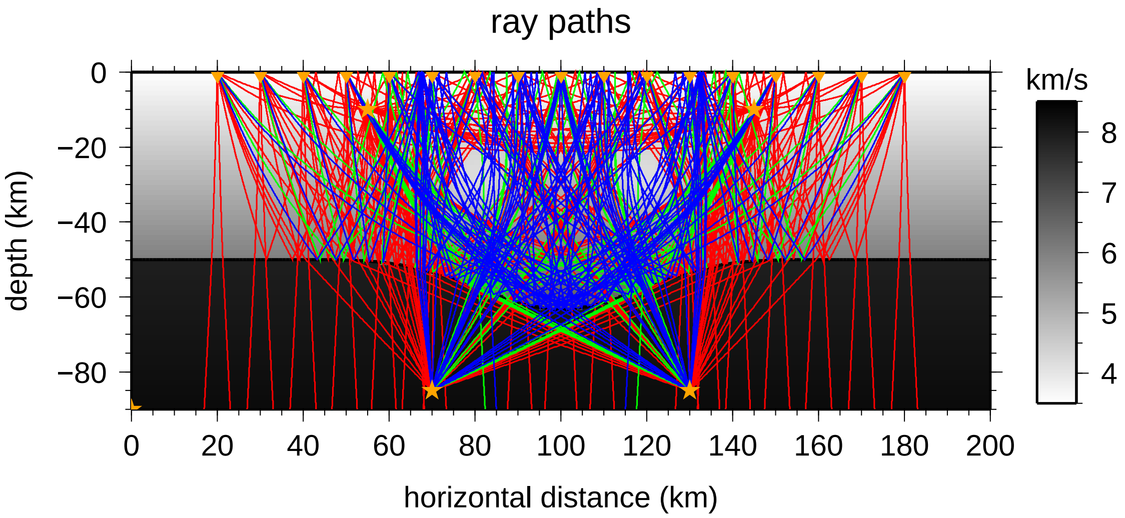


Figure 5.10: Layered velocity model and associated paths used in the second inversion test involving later arrivals. Ray paths corresponding to the first, second and remaining later arrivals are plotted as red, green and blue lines, respectively. Point sources are denoted by stars and receivers by triangles. Note that paths also emanate from two impinging plane wavefronts. The model perturbations, to be recovered are more clearly observed in figure 5.11.

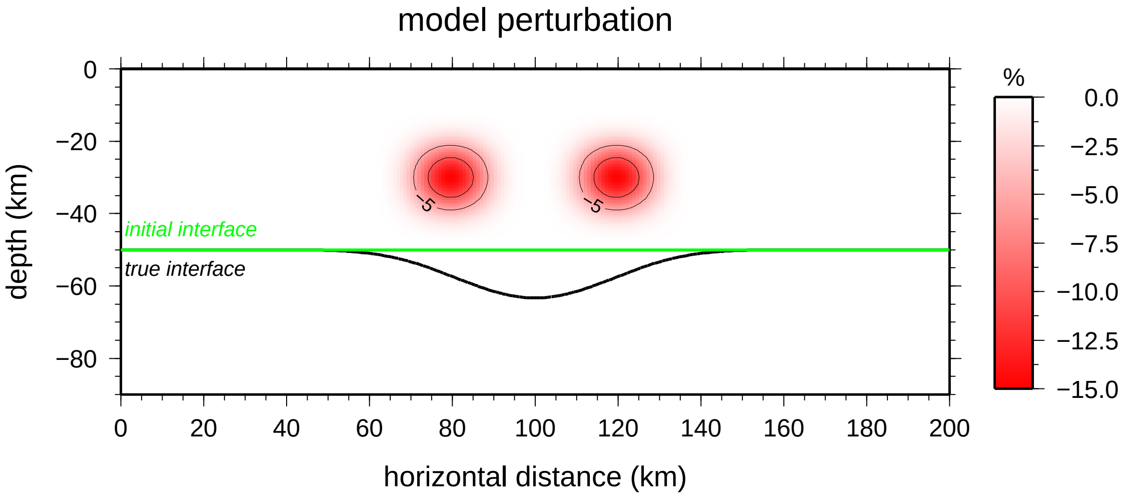


Figure 5.11: The interface perturbation and velocity anomalies that define the true model. The contour lines are plotted at 5 % intervals. In figure 5.10 the two velocity anomalies are masked by the ray paths.

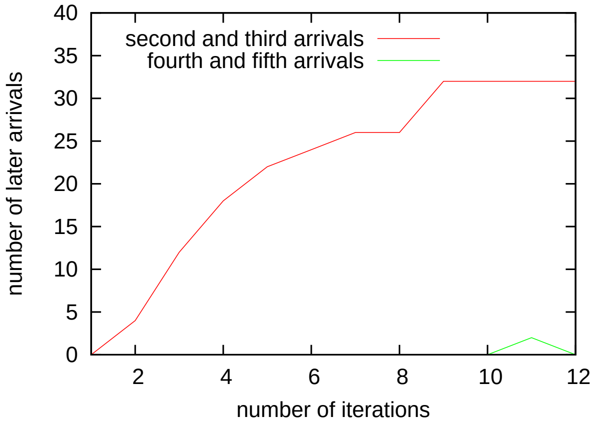


Figure 5.12: Number of later arrivals as a function of iteration when only first arrivals are used during the iterative inversion procedure.

is used to compute $\delta \mathbf{m}$ and the iterative inversion procedure is stopped if the RMS error of the travel time residuals falls below 0.02 s. The subspace dimension is automatically adapted if the space spanned by the basis vectors is less than 12 dimensions.

When only first arrivals are used in the inversion, the final model does not generate any fourth, fifth or sixth arrivals. In figure 5.12, second and third arrivals appear gradually up to the ninth step of the iterative inversion procedure. After the ninth iteration, the number of second and third arrivals is constant (32). It is interesting to note that the model used at iteration 11 generates two fourth and fifth arrivals. Figure 5.13 shows the number of later arrivals as a function of iteration when first and later arrivals are used in the inversion. Second and third arrivals appear after the first iteration, and once the perturbations are large enough, fourth, fifth and sixth arrivals start to appear. Similar to the previous example (section 5.4.1), the introduction or removal of later arrivals tends to be gradual rather than dramatic. The 60 predicted second and third arrivals are close to the observed number of 64. The 18 predicted fourth and fifth arrivals is also close to the observed number of 20; the two observed sixth arrivals are also predicted.

For a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory and a GNU/Linux operating system, the computation time for the iterative inversion procedure when only first arrivals are used is 38 min; this compares to 105 min when first and later arrivals are used. The multi arrival tomography takes significantly longer because of the larger number of iterations and the greater number of arrivals for which

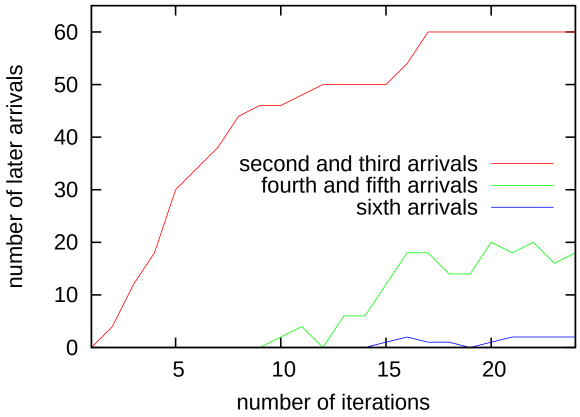


Figure 5.13: Number of later arrivals when first and later arrivals are used during the iterative inversion procedure (cf. figure 5.12).

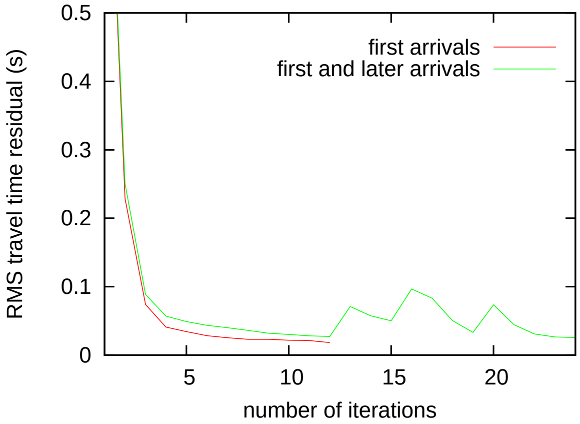


Figure 5.14: RMS travel time residuals for the two solutions.

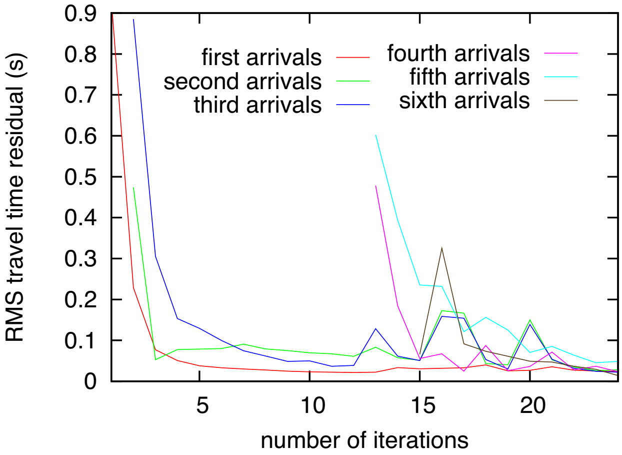


Figure 5.15: RMS travel time residuals for the different arrivals when first and later arrivals are used in the iterative inversion procedure.

ray paths and Fréchet derivatives have to be computed. The computation times in this example are significantly larger than those for the smooth velocity model (section 5.4.1). Recall that the version of the Lagrangian solver which can propagate wavefronts across interfaces relies on a scratch file on disk to store the wavefronts so that they can be used to track rays back to the source after the wavefront propagation is finished. For the smooth velocity model wavefronts are kept in memory, which makes the process significantly faster.

Figure 5.14 shows the RMS travel time residuals or error for the two solution classes. When only first arrivals are used, the RMS error decreases monotonically. On the other hand, when first and later arrivals are used, the RMS error initially decreases, then increases as fourth, fifth and sixth arrivals appear, before undulating for a few iterations and finally decreasing. This behaviour is due to the appearance of fourth, fifth and sixth arrivals, to which the structure has yet to be adjusted. In figure 5.15 the RMS travel time residuals for the different arrivals are plotted. It is important to note that the RMS error of the first arrivals decreases within the first 10 iterations and then stays relatively constant. Also in the first 10 iterations, the RMS error of the second arrivals decreases initially but then stays at a relatively large value. For the third arrivals the RMS error also decreases in the first 10 iterations but is smaller than for the second arrivals. When the fourth, fifth and sixth arrivals are predicted for the first time they show relatively large travel time residuals. As they are taken into account during later steps of the iterative inversion procedure, the RMS error for second and third arrivals increases. One gets the impression that adjusting the structure for one class of later arrivals can increase the misfit for another class. This behaviour of the RMS error could also be due to a mismatch between the observed and predicted later arrivals for several source-receiver pairs (i.e. when the number of predicted arrivals is not equal to the number of observed arrivals, as discussed earlier). Only after the 20th step does the RMS error decrease for all later arrivals. Note that in figure 5.13 fourth and fifth arrivals are predicted at the 10th and 11th iteration. They, however, do not appear in the plot of the RMS error (figure 5.15) of the different arrivals, due to the fact that there are no observations for these later arrivals.

Figure 5.16 shows the difference between the inversion result and the true structure when only first arrivals are used in the inversion. The trade off between interface geometry and velocity anomaly is clearly not as well resolved compared to when both first and later arrivals are used (figure 5.17). If first arrivals are used, it takes 12 iterations to reduce the RMS error of the travel time residual to less

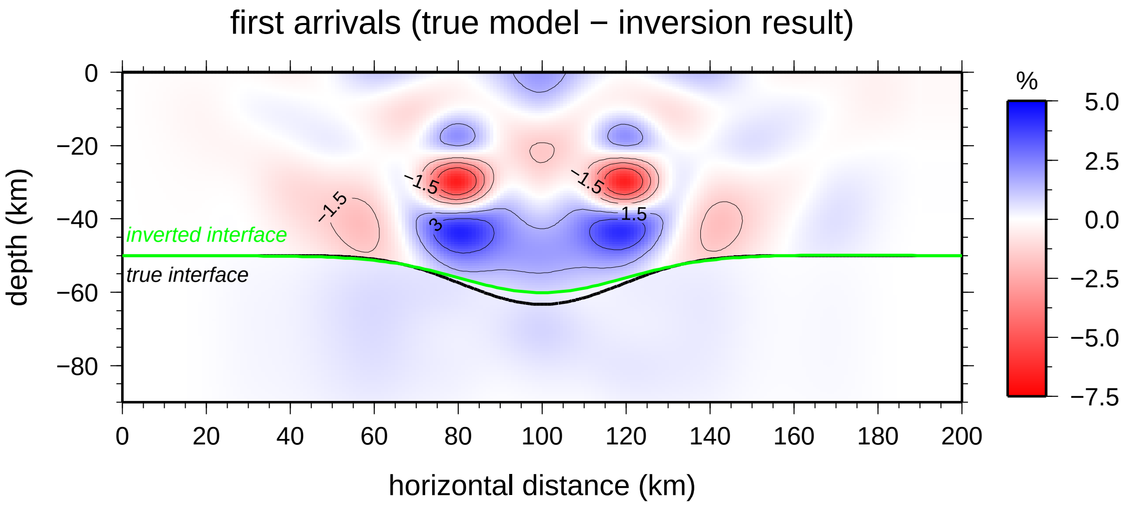


Figure 5.16: Relative error in percent between the inversion result based on first arrivals and the true model. The contour lines are plotted at 1.5 % intervals.

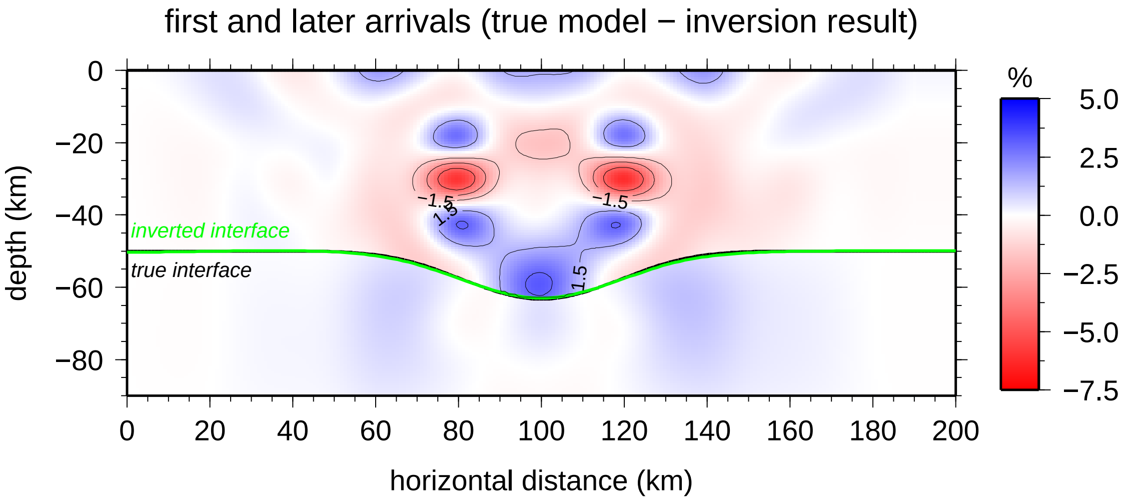


Figure 5.17: Relative error in percent between the inversion result based on first and later arrivals and the true model. The contour lines are plotted at 1.5 % intervals.

than 0.02 s, compared with 24 iterations, when first and later arrivals are used.

In general it turns out that adding later arrivals makes the inverse problem much more non-linear, which means that an iterative non-linear approach may fail. In such a situation, using first arrivals only may result in a better solution. This possibility is investigated by reducing the number of unknowns to only two parameters: (1) the amplitude of the two low velocity anomalies (i.e. the dimensionless factor λ by which the original velocity value is multiplied to obtain the perturbation) and (2) the depth of the central interface node (see figure 5.18a). Consequently, model space is now only two dimensional. The same synthetic model as before (see figure 5.10) is used but the velocity perturbation, whose recovery is attempted using different sets of arrivals, has a slightly larger amplitude. This leads to a larger number of observed later arrivals for this test case, as shown in table 5.2.

first arrivals	170	fourth arrivals	26
second arrivals	46	fifth arrivals	26
third arrivals	46	sixth arrivals	16
total (1st, 2nd and 3rd)	262	total (4th, 5th and 6th)	68

Table 5.2: Table showing the number of observed arrivals for the simplified model, defined using two model parameters: the amplitude of the two low velocity anomalies and the depth of the central interface node.

Figure 5.18b shows the number of predicted second and third arrivals, and the number of predicted fourth, fifth and sixth arrivals as a function of the two unknowns. The position of the starting model is marked by the green arrow and the red arrow marks the true model, which is to be recovered by the inversion scheme. The starting model does not generate any later arrivals, and only once the model perturbation is large enough do second and third arrivals appear. Perturbations need to be even greater to observe fourth, fifth and sixth arrivals. Another notable observation is that only in the vicinity of the true model do the predicted number of later arrivals correspond to the observed number.

Figure 5.19 shows the travel time misfit (RMS error) as a function of position in model space using: (1) first arrivals (figure 5.19a), (2) first, second and third arrivals (figure 5.19b) and (3) all arrivals (figure 5.19c). In an iterative inversion procedure the global minimum of such a misfit function is targeted using local gradient information (i.e. a downhill search towards a minimum). The method does not search uphill and hence there is no guarantee that the located minimum

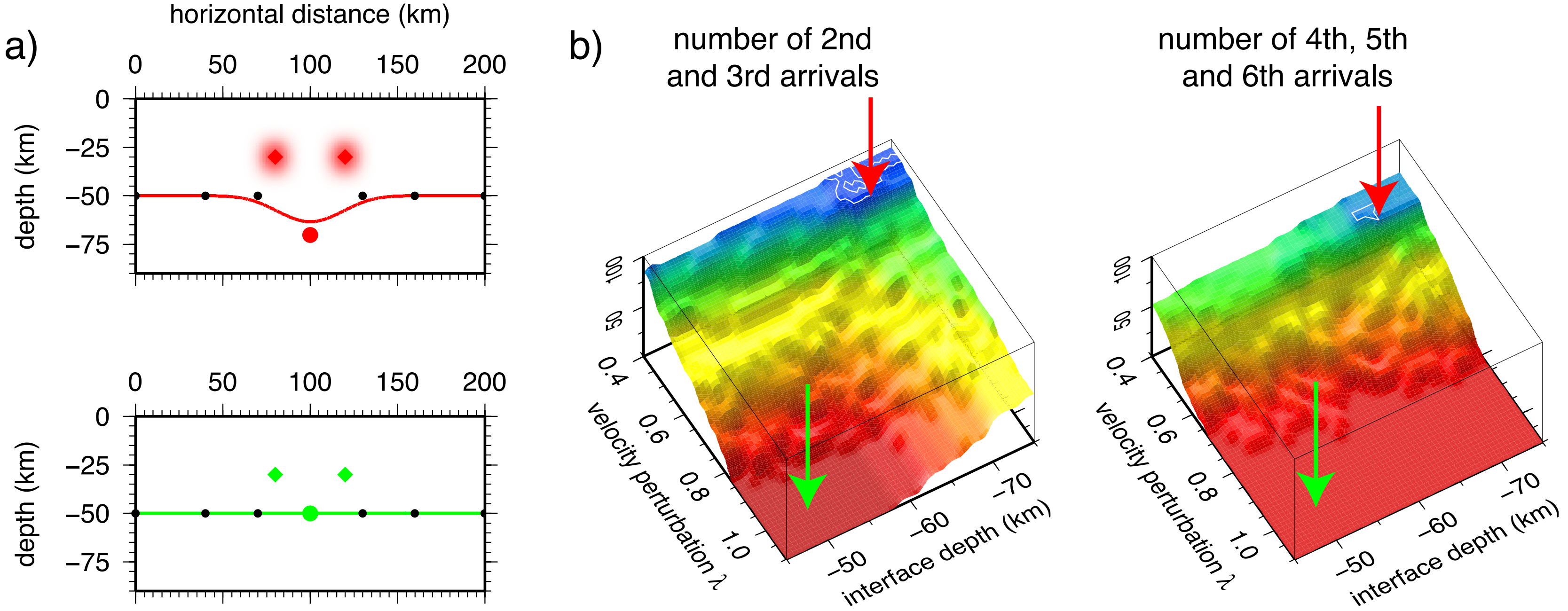


Figure 5.18: (a) True model (top) and starting model (bottom); (b) number of arrivals as a function of the two unknowns of the simplified model. The starting model is marked by a green arrow and the true model is marked by a red arrow in figure (b). The white contour lines denote the number of arrivals in the true model.

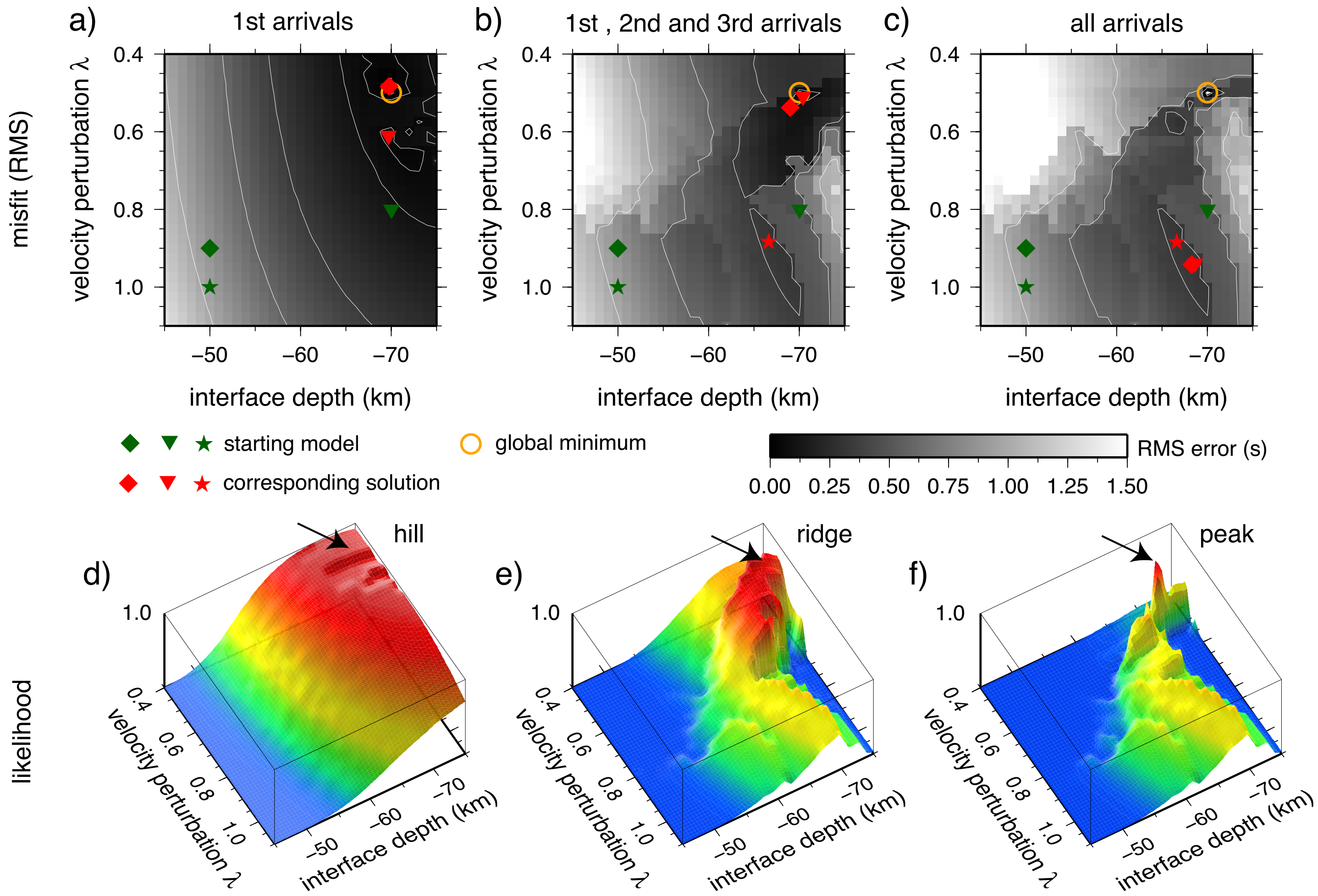


Figure 5.19: Top row: travel time misfit (RMS error) as a function of the two unknowns (figure 5.18) if (a) 1st arrivals (b) 1st, 2nd and 3rd arrivals (c) all arrivals are used. The true solution at the global minimum (0.0) is marked by an orange circle. The symbols in (a), (b) and (c) denote different starting models and the corresponding solution for a conjugate gradient method. Bottom row: likelihood computed as a function of the two unknowns if (d) 1st arrivals (e) 1st 2nd and 3rd arrivals (f) all arrivals are used. The true solution (1.0) is marked by a black arrow.

is a global minimum.

To illustrate how the non-linearity of the multi arrival tomography problem affects the ability of a linearised approach to find a global minimum, a conjugate gradient method is applied for the different misfit functions (e.g. Press et al., 1992). Different starting models are indicated by the green symbols in figures 5.19a, b, c. The corresponding solutions are denoted by the same symbol in red. If only first arrivals are used, all three starting models lead to solutions close to the global minimum. If first, second and third arrivals are used a local minimum is found for one of the starting model. Finally, if all arrivals are used, none of the starting models result in solutions that converge to the global minimum. This is because the global minimum of the misfit function is at the end of a valley behind a small ridge (figure 5.19c). The conjugate gradient method would need to search uphill in order to reach the global minimum, something which is not possible for standard gradient based methods.

A likelihood function can be defined to better illustrate the non-linearity of the inverse problem. The likelihood L is given by

$$L = \exp\left(-\frac{1}{2}\sum_{i=1}^{n}\frac{(d_i - t_i(m))}{\sigma_i^2}\right), \tag{5.23}$$

where d_i is the observed travel time, $t_i(m)$ the model prediction, σ_i^2 the standard deviation of the observed travel time (assumed to be 0.3 s for this test) and n the number of observations. A likelihood of 1.0 corresponds to a perfect fit between observation and model prediction. In the case of the misfit function, the true solution is at the global minimum, while for the likelihood function it is at the global maximum. From the likelihood values in figure 5.19d it is clear that in the first arrival case, the global maximum is at the summit of a hill, but the actual peak is not very well defined. If first, second and third arrivals are used, the hill becomes a ridge containing several small distinct peaks, of which one corresponds to the global maximum (figure 5.19e). If all arrivals are used the global maximum is revealed as a sharp peak, but there are several surrounding local maxima on top of which a gradient based method could easily become trapped (figure 5.19f). This is in fact what happens when a conjugate gradient method is applied to invert all arrivals (see figure 5.19c).

The above example demonstrates that later arrivals contain additional structural information. They make the minimum of the misfit function and the maximum of the likelihood function much more well defined. However, they also increase the non-linearity of the misfit function, which can divert a gradient based search algorithm into a local extremum and hence result in an undesirable solution. A fully non-linear approach (e.g. Gill et al., 1981; Kirkpatrick et al., 1983; Whitley, 1994) would have little problem in finding the global maximum of the likelihood function if all arrivals are used, and it would actually benefit from the global maximum being much better constrained than in the first arrival case. The sharpness of the peak of the likelihood function in figure 5.19f suggests that errors in data will map into a smaller region around the solution. It remains to be seen whether or not a global search algorithm would be feasible for a tomographic problem with several hundred model parameters or more. An alternative would be to only use later arrivals once the solution model is in the vicinity of the true model. However, there is no guarantee that the first arrivals could bring us sufficiently close to the true model to enable the successful inclusion of later arrivals. This is similar to the situation in full waveform inversion, where first arrival tomography is used to generate a sufficiently accurate starting model to validate the assumption of local linearity (e.g. Brenders & Pratt, 2006).

5.5 Summary

In this chapter, synthetic tests have shown that later arrivals clearly contain additional information about velocity and interface structure, and that they can be successfully exploited in seismic tomography. However, the inclusion of later arrivals increases the non-linearity of the inverse problem, thus making it less amenable to solution using gradient based methods. To help mitigate this effect, first arrival inversion can be applied to localise the solution region prior to application of multi arrival tomography. Alternatively, a fully non-linear inversion could be used provided the number of unknowns is not too large. If an iterative non-linear inversion process is used, one should carefully monitor the RMS errors for different arrivals in order to determine when it is best to stop updating the model. Despite these difficulties, in both numerical tests (i.e. the smooth velocity model and the layered velocity model) an iterative non-linear approach was successfully used, and structure was more accurately recovered when later arrivals were included compared to when only first arrivals were used.

Figure 5.20 shows the relative locations of different seismic imaging techniques on an illustrative plot which maps out their use of arrival and frequency informa-

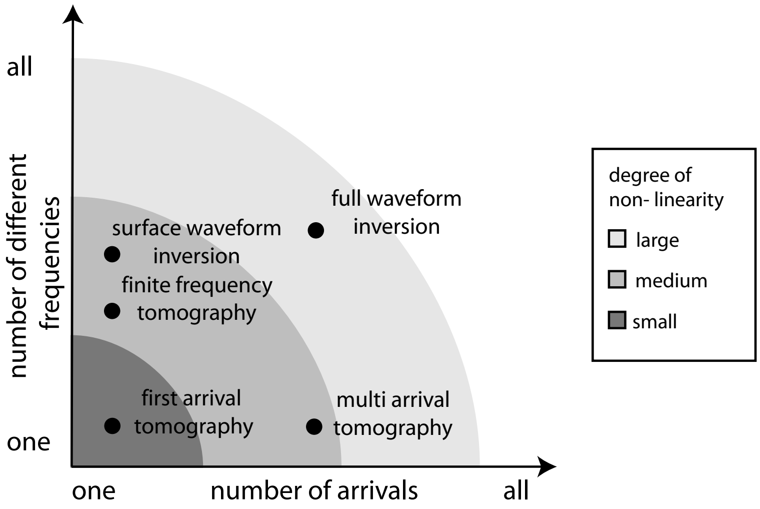


Figure 5.20: Cartoon showing the relative location and non-linearity of different imaging schemes with respect to their use of arrival information and frequency.

tion. First arrival tomography is located in the lower left corner because it uses only a small part of the recorded wavetrain. Full waveform inversion potentially uses all arrivals and all available frequencies contained in an observed seismogram and is therefore located in the upper right corner. Finite frequency tomography uses Fréchet kernels for finite frequency body waves (i.e. Banana-doughnut kernels). While it only exploits the first arrivals, it does so at different frequencies (e.g. Dahlen et al., 2000). The concept of multi arrival tomography uses all arrival but is based on ray theory. However since the computation of Fréchet kernels for finite frequency travel times still requires the location of geometrical ray paths, one could potentially use the additional ray paths provided by later arrivals in the Fréchet kernels for finite frequency tomography. Such a combination would then allow even more of the recorded seismogram to be used.

The next logical step is to apply the method to real data. The main obstacle is the correct identification of later arrivals in an observed seismogram. The Lagrangian solver used in this work can calculate the travel time and the direction of the incoming ray for each arrival at a receiver. Using Gaussian beams one can also compute synthetic seismograms for body waves. Considering that multipathing of surface waves is a frequently observe phenomenon (e.g. Capon, 1971), the most feasible step might be the use of a multi arrival tomography scheme in surface wave tomography, or possibly even for ambient noise tomography. At this stage, a surface wave or ambient noise dataset with observed multipathing is needed to explore the potential of multi-arrival tomography in real data applications.

Chapter 6

Multi valued travel times in three dimensions

The Eulerian and Lagrangian schemes presented earlier can both be extended to three dimensions. In this case the wavefront is tracked in six dimensional phase space or five dimensional reduced phase space. In the Eulerian scheme, the bicharacteristic strip becomes a two dimensional manifold (i.e. a surface) in five dimensional reduced phase space and is defined as the intersection of three four dimensional manifolds (Osher et al., 2002). Each of the three four dimensional manifolds are evolved using the level set method. It was discussed in chapter 2 how the finite difference scheme for the level set equation can be used for the general case of an evolving $m-1$ dimensional manifold in an m dimensional space using a dimension by dimension approach. The Eulerian scheme requires three five dimensional problems to be solved in order to find the solution to a three dimensional problem. Increasing the grid resolution by a factor of two leads to an increase in computation time by a factor of 64 and memory requirements by a factor of 32. An efficient implementation of an Eulerian scheme with a local level set method and adaptive gridding is therefore necessary. Given the complexity of implementing such an advanced algorithm, and the associated computational cost, it is much more feasible to use a Lagrangian solver and define the wavefront using a triangular topology. Triangles and nodes can then be added or removed based on the phase space distance between the nodes.

Several Lagrangian schemes for wavefront tracking in three dimensions using triangulated wavefronts have previously been proposed. Vinje et al. (1999) describe the wavefront in real space and refine it based on the distance between the

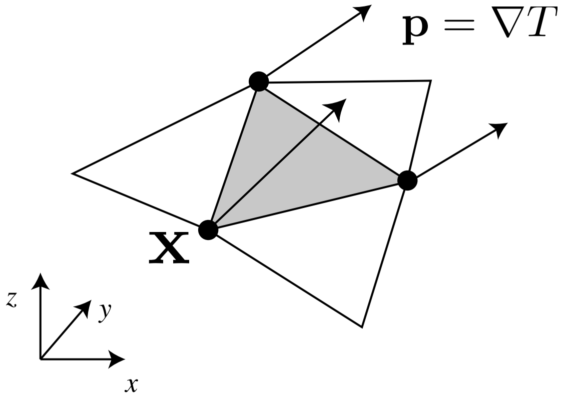


Figure 6.1: Part of a wavefront in three dimensional real space. The wavefront is triangulated and for each node on the wavefront (solid black dot) the position $\mathbf{x}$, the slowness vector $\mathbf{p}$ and connectivity information is stored.

nodes and the angle between the wavefront normals. Lucio et al. (1996) use their Hamiltonian formulation of ray theory (Lambaré et al., 1996) for wavefront tracking in three dimensions. More recently Lee & Gibson Jr. (2007) discuss the use of cubed-sphere meshes to describe the wavefront. In contrast to previous schemes, algorithms first proposed in the field of computer graphics for the description of surfaces, are used to describe the wavefront in this chapter. The novel approach taken in this work also differs from previous schemes in that nodes are not only added to the wavefront as it evolves, but also removed if they are no longer necessary to describe the wavefront. Removing nodes has the potential to reduce computation time as only the minimum number of nodes necessary to describe a wavefront with the required accuracy is used, so no local over sampling of the wavefront occurs.

6.1 Representing the wavefront

For problems in two dimensional real space, the wavefront is a line (i.e. a one dimensional manifold). Discretising the bicharacteristic strip in this case is relatively straightforward; it can be done with a linked list where each point knows its two neighbours. For three dimensional models, the wavefront and bicharacteristic strip are surfaces (i.e. a two dimensional manifold). Figure 6.1 illustrates how the wavefront is described using a triangular mesh. For each vertex of the wavefront, its position $\mathbf{x}$ and slowness vector $\mathbf{p}$ are stored. The triangulation defines the connectivity between the set of vertices describing the wavefront. Each vertex knows its neighbours and to which triangle it belongs. Similarly, each triangle knows its neighbours and its constituent nodes. This connectivity information is particularly important when triangles are added or removed from the wavefront. The data structure for the wavefront in this work is based on pointers, as they allow fast access to topological information such as edges or neighbouring faces of the triangles.

The state of a node on the wavefront (see figure 6.1) is characterised by its position $\mathbf{x}$ and its motion given by the slowness vector $\mathbf{p}$ (see section 2.3). The wavefront can then be propagated in time using the Hamiltonian formulation of ray theory introduced in section 2.3,

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = c\frac{\mathbf{p}}{|\mathbf{p}|}, \tag{6.1}$$

$$\frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} = -|\mathbf{p}|\nabla c, \tag{6.2}$$

where $\mathbf{p} = \nabla T$ is the slowness vector, $\mathbf{x}$ the position vector in three dimensional real space, c the speed and t the time. Smooth variations in wave speed $c(x, y, z)$ are defined by a mosaic of cubic B-spline volume elements, the values of which are determined by a regular mesh of control vertices (see appendix B).

For a particular time step, the position and slowness vector of each node on the wavefront is first updated using a fourth order Runge Kutta method (appendix C). Nodes (and hence triangles) are then added and removed based on the distance in phase space between nodes. Details about the refinement and simplification strategies used in the new scheme are provided in the next two sections.

The concept of phase space and reduced phase space was introduced for two dimensional real space in section 2.3. When tracking wavefronts in three dimensional models, reduced phase space is five dimensional, with the additional dimensions provided by the azimuth and inclination of the direction vector. The azimuth range is bounded between $-\pi$ and π and the inclination spans $-\pi/2$ to $+\pi/2$. However, from an implementation point of view it is more convenient to use full phase space because periodicity does not need to be considered when interpolating slowness vectors for newly inserted nodes. Therefore a full phase space approach will be used for the extension of the Lagrangian solver to three dimensions and hence the bicharacteristic strip will lie in six dimensional space.

A three dimensional space spanned by the three moment variables, (i.e. the components of the slowness vector) can be defined. The initial bicharacteristic

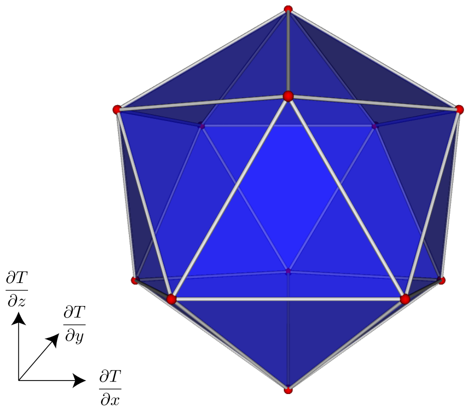

$\frac{\partial T}{\partial z}$
$\frac{\partial T}{\partial y}$
$\frac{\partial T}{\partial x}$

Figure 6.2: Icosahedron used to describe the point source. The three dimensions in this figure are the three components of the slowness vector. For a point source all nodes have the same position **x** but different slowness vectors **p**. Thus their position in six dimensional full phase space is not the same.

strip for a point source corresponds in this slowness space to a sphere. However, in practice an icosahedron is used to initialise the nodes and triangles describing the bicharacteristic strip for a point source. An icosahedron provides uniform coverage of a sphere using 12 points with all nodes having the same number of neighbours, and all faces having the same shape and size. Thus, there is no bias in the initial distribution of points, which might otherwise arise if a sphere gridded in latitude and longitude was used. It will be shown in the next section that an icosahedron can be subdivided to produce a sphere, given an appropriate refinement scheme. Subdivision of an icosahedron to describe a sphere is a common approach. For example, Wang & Dahlen (1995) use it in their development of a spherical cubic B-spline parameterization for a two dimensional function on a sphere. When it comes to implementation, setting up the node and triangle associations for an icosahedron is more convenient than for a large number of points distributed over a sphere. The points of the initial icosahedron have the same position **x** but different slowness vectors **p** (i.e. each node will propagate away form the source in a different direction). While it is not possible to visualise a six dimensional phase space, it is possible to visualise a subset, as demonstrates in figure 6.2. Vinje et al. (1996a) also use an icosahedron to initialise their point source in real space.

The point source in two dimensional problems was represented with up to 150 points on the bicharacteristic strip (section 3.2), so 12 points are unlikely to be sufficient to represent the point source in three dimensions. Therefore, the initial wavefront given by the icosahedron needs to be refined by inserting nodes prior to wavefront propagation. Fortunately, it is possible to obtain a sphere by refining the icosahedron using a higher order interpolation scheme.

Algorithms for surface refinement have been available for more than 25 years (e.g. Catmull & Clark, 1978) and used in computer graphics for computer aided design, and more recently movie production and computer games. The simplification of a surface can be as important as its refinement, and is motivated by the goal to describe an object with a given accuracy using the least number of polygons (e.g. Garland & Heckbert, 1997). There exists a wide variety of interpolation and approximation schemes for surface refinement and simplification. From a wavefront tracking point of view, refinement and simplification algorithms which act locally and allow for self-intersecting surfaces are of great relevance. Refinement and simplification algorithms which re-triangulate the entire surface, even when only one triangle is added or removed, are not efficient in this context.

Explicit techniques for parameterizing evolving surfaces that have been developed in the field of computer graphics have seen limited application in the Earth sciences, although Schmalzl & Loddoch (2003) discuss surface simplification and refinement strategies in their modelling of chemical heterogeneities in geophysical flows. However, the difference between their chemical boundary and a seismic wavefront is that the latter can become self-intersecting in complex media.

6.1.1 Surface refinement

The concept of subdivision was introduced for a line when discussing a higher order interpolation method for the Lagrangian scheme in section 3.1. If subdivision is applied to a surface, the geometry of the initial mesh is important. In a triangular mesh, each vertex is typically connected to 6 neighbours. One typically refers to the number of neighbours of a node as its valence. Figure 6.3a shows a triangular mesh with new vertices added along the edges. Note that all vertices, which are not located on the boundary of the grid, have a valence of 6 (i.e. six neighbouring nodes). Figure 6.3b shows an initial triangulation where the node in the middle has a valence of 4. For a triangular mesh one refers to vertices with a valence not equal to 6 as extraordinary vertices.

Figure 6.4 illustrates an edge split applied to a mesh with arbitrary topology. Let us assume that the triangle *abc* has been marked for refinement by some

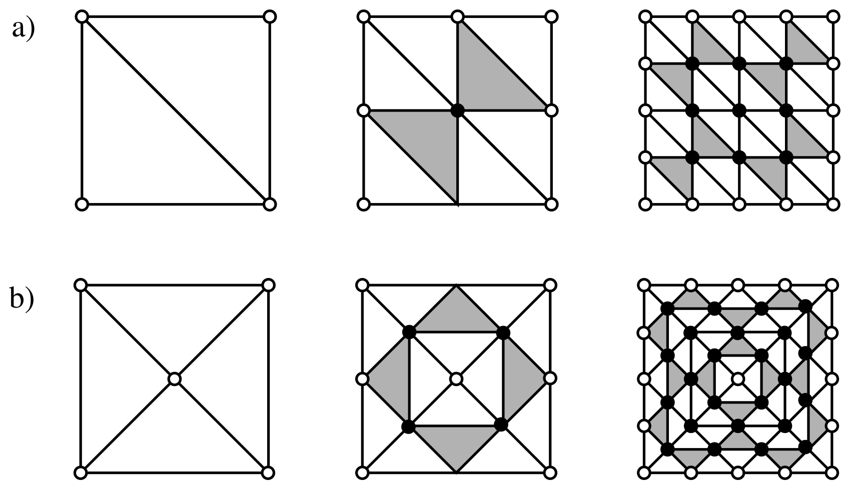


Figure 6.3: Triangular subdivision scheme applied to two different initial triangulations. New vertices are added along the edges. Newly added triangles are grey, nodes with a valence of 6 are black and those with a valence not equal to 6 are white. (a) All vertices, except those on the boundary of the domain, have a valence of 6. (b) An extraordinary vertex with a valence of 4 in the centre of the mesh.

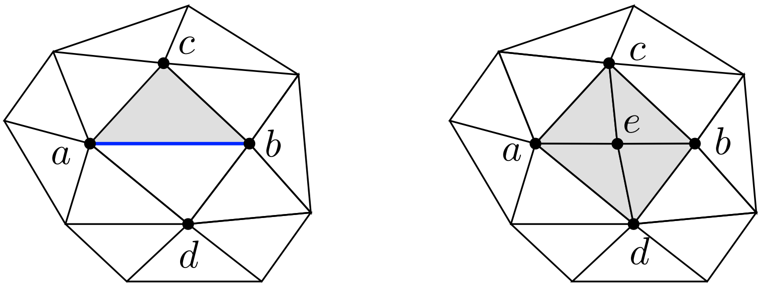


Figure 6.4: Demonstration of an edge split. The edge ab along which the grey triangle is split is marked in blue.

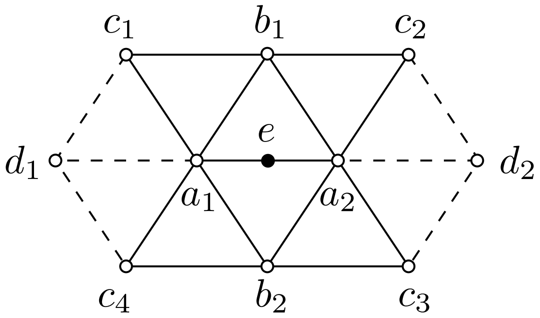


Figure 6.5: Butterfly scheme with solid lines indicating the eight point stencil. Additional points for the ten point stencil of the modified butterfly scheme are marked by dashed lines.

criteria that is based on the properties of the triangle, and that one of the edges has been chosen for edge splitting. A point e is inserted along this edge and the triangle abc and its neighbour adjacent to the edge ab are each subdivided into two triangles. The four newly formed triangles aec, ebc dbe and dea replace the old triangles abc and adb. The position of the newly added point e can be calculated using any interpolation or approximation scheme based on the nodes a and b and their neighbours. If the neighbours of a and b are used in the interpolation or approximation method, the fact that b is an extraordinary vertex has to be considered in the scheme.

Subdivision schemes are either approximating or interpolating. In an approximating subdivision scheme the original points are removed while in an interpolating scheme the original points are part of the refined set (as in figure 6.4). When adding triangles to the wavefront the position of pre-existing nodes should be preserved, therefore an interpolating subdivision scheme is needed. The butterfly scheme (Dyn et al., 1990a) is locally interpolating and leads to continuous surfaces. It was originally defined using an eight point stencil (Dyn et al., 1990a) which was later extended to a ten point stencil (Dyn & Levin, 1994). The position of a new vertex is computed using a weighted sum over nodes of the stencil shown in figure 6.5. The weights for the nodes are as follows

$$a_{1,2} = \frac{1}{2} - w,\ b_{1,2} = \frac{1}{8} + 2w,\ c_{1,2,3,4} = -\frac{1}{16} - w,\ d_{1,2} = w. \tag{6.3}$$

The tension parameter w determines how tightly the surface is pulled towards the control net. For $w = 0$ the original eight point stencil is obtained. The position $\mathbf{x}$ of the new node e in figure 6.5 is given by

$$\begin{aligned}\mathbf{x} = &\left(\frac{1}{2} - w\right)(a_1 + a_2) + \left(\frac{1}{8} + 2w\right)(b_1 + b_2) \\ &- \left(\frac{1}{16} + w\right)(c_1 + c_2 + c_3 + c_4) + w\,(d_1 + d_2)\,,\end{aligned} \tag{6.4}$$

where a_1, a_2, b_1, ... represent the position of the corresponding vertices. Similarly, the slowness vector $\mathbf{p}$ for the new node e can be interpolated.

Dyn et al. (1990b) show that the butterfly scheme based on the eight point stencils leads to surfaces with continuous first derivatives if all vertices of the underlying grid have a valence of 6. However, when dealing with extraordinary vertices, i.e. vertices with a valence other then six, it is no longer obvious which

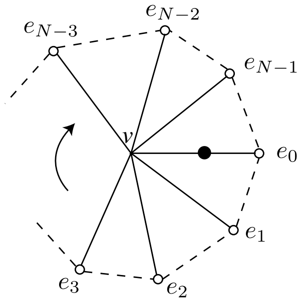


Figure 6.6: Stencil for an extraordinary vertex v in the modified butterfly scheme. The new vertex to be computed is denoted by the black circle.

vertices should be used for the interpolation of the new node. Zorin et al. (1996) therefore extend the original butterfly scheme to allow it to handle vertices of arbitrary valence. The modified butterfly scheme distinguishes between three cases based on the valence of the nodes a and b (see figure 6.4).

1. *The edge connects two vertices of valence 6:* In this case an eight point stencil with weights given by (6.3) is used and w set to zero.

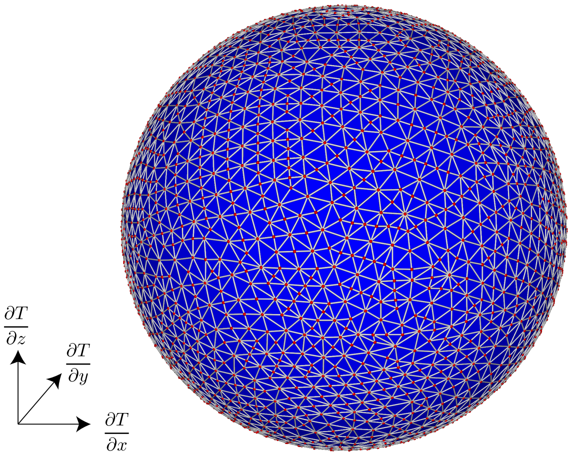
2. *The edge connects a vertex of valence six with a vertex of valence N, where $N \neq 6$:* Neighbours of the vertex with a valence of N are used in the stencil (figure 6.6). For the different valences N the weights are:

$$\begin{aligned} N = 3 : v &= \frac{3}{4},\, e_0 = \frac{5}{12},\, e_1 = -\frac{1}{12},\, e_2 = -\frac{1}{12}, \\ N = 4 : v &= \frac{3}{4},\, e_0 = \frac{3}{8},\, e_1 = 0,\, e_2 = -\frac{1}{8},\, e_3 = 0, \\ N \geq 5, N \neq 6 : v &= \frac{3}{4},\, e_j = \frac{1}{N}\left(\frac{1}{4} + \cos\left(\frac{2\pi j}{N}\right) + \frac{1}{2}\cos\left(\frac{4\pi j}{N}\right)\right), \qquad (6.5) \end{aligned}$$

   with $j = 0, 1..., N-1$. This case corresponds to the example in figure 6.4, where the node a has a valence of 6 and the node b a valence of 5.

3. *The edge connects two extraordinary vertices*: The average value of the two points computed separately by applying (6.5) to each extraordinary vertex is taken to be the new interpolated node.

The modified butterfly scheme is used in this work not only to compute the position $\mathbf{x}$ of the newly inserted points but also to calculate their slowness vector $\mathbf{p}$. For computational convenience, only triangles which have neighbours across all three edges are considered for refinement. This means that triangles on the wavefront near the boundary of the computational domain are not refined.



$\frac{\partial T}{\partial z}$
$\frac{\partial T}{\partial y}$
$\frac{\partial T}{\partial x}$

Figure 6.7: Refinement of the initial icosahedron used to describe the point source. The three dimensions in this figure are the three components of the slowness vector associated with each point. For a point source, all nodes have the same position **x** but different slowness vectors **p** (i.e. different initial directions), which means that their position in six dimensional full phase space is not the same.

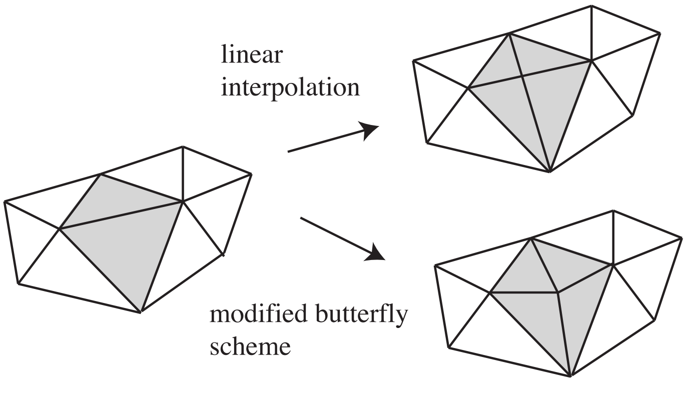


Figure 6.8: The shape of the resulting triangles when an edge split is applied depends on the interpolation scheme. The modified butterfly scheme leads to a more balanced triangulation as far as the area of the individual triangles is concerned.

Similar to the situation with a hexagon in two dimensions, where a circle is obtained by successive application of a subdivision scheme (see section 3.1), a sphere is produced by applying the modified butterfly scheme several times to the initial icosahedron (figure 6.7). This allows the point source to be initialised with an icosahedron and then refined using edge splits until the required point density on the bicharacteristic strip is achieved. The mesh describing the sphere is no longer given by equilateral triangles. This is a consequence of using the modified Butterfly scheme (i.e. edge split) instead of splitting every triangle into four equilateral triangles (i.e. triangular subdivision), which would lead to a regular mesh as illustrated in figure 6.3. Triangular subdivision works well if the triangle to be subdivided is equilateral as well. The reason why the modified Butterfly scheme is used in this work instead of triangular subdivision is that we expect our triangles to no longer be equilateral once the wavefront begins to propagate.

Unlike linear interpolation, the modified butterfly scheme has the feature that newly inserted triangles do not have to lie on the existing edge of the previous triangle, which is an advantage for smooth surfaces. This is illustrated in figure 6.8, where the area spanned by the two triangles is replaced with four triangles. The modified butterfly scheme therefore may lead to a more balanced triangulation than linear interpolation.

Vinje et al. (1996b) also perform an edge split when adding new rays to the wavefront. However, they do not use a unified metric in phase space to decide when to perform an edge split. They refine the wavefront if the distance between two nodes in real space or the angle between the corresponding wavefront normals, is above a certain threshold. To determine the position of the new node, they fit a curve (3rd order polynomial) through the points a and b using the direction of the wavefront normals at a and b as well as wavefront curvature information obtained by dynamic ray tracing. In the modified butterfly scheme, the wavefront curvature is taken into account by using the neighbouring nodes of a and b in the formulation of the interpolation stencil. Therefore, in the scheme introduced here, only the kinematic ray tracing equations have to be solved during wavefront tracking.

6.1.2 Surface simplification

The idea behind surface simplification is to ensure that the wavefront is described by the minimum number of nodes and triangles necessary to achieve a given accu-

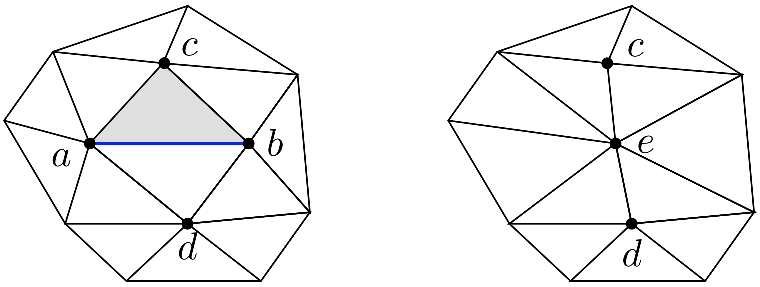


Figure 6.9: Edge contraction: the edge along which the grey triangle is contracted is marked in blue.

racy. Any triangle that contains a node which has left the computational domain is removed from the wavefront. As a result there may be gaps between the wavefront and the boundary of the computational domain. Therefore, the computational domain is chosen to be slightly larger than the region of interest.

In general, the surface area of a propagating wavefront increases as time progresses. Locally however, it is possible that triangles may shrink due to wavefront distortion caused by the velocity field. Removing triangles which are not necessary to describe the wavefront within a given tolerance therefore helps to minimise the overall computation time and memory requirements.

Garland & Heckbert (1997) discuss surface simplification using quadric error metrics. They quantify the error caused by removing a vertex from the grid, and only remove the vertex if the error is below a certain threshold. They also allow unconnected sections of the surface to be joined. For multipathing wavefronts unconnected segments should not be joined by surface simplification. Therefore only edge contractions are performed.

The edge contraction used to remove triangles and nodes is similar to the edge split used to insert triangles and nodes. Figure 6.9 illustrates an edge contraction where the two nodes a and b are replaced by a new node e. As a consequence, the two triangles abc and adb are also removed. The position and direction of the slowness vector for the new node e can be computed using linear interpolation or the modified butterfly scheme. Following the edge contraction the shape of the triangles, which contain a node that has been removed (i.e. replaced by a new node), is changed. These triangles must now also span the region previously spanned by the two triangles which have been removed.

6.1.3 Mesh quality

A unified distance in phase space σ between two nodes along the edge of a triangle is defined as follows:

$$\sigma = \Bigg[\left(\frac{\Delta x_1}{l}\right)^2 + \left(\frac{\Delta x_2}{l}\right)^2 + \left(\frac{\Delta x_3}{l}\right)^2 + \left(\frac{\Delta p_1}{s_{max}}\right)^2 + \left(\frac{\Delta p_2}{s_{max}}\right)^2 + \left(\frac{\Delta p_3}{s_{max}}\right)^2 \Bigg]^{\frac{1}{2}} \tag{6.6}$$

where $s_{max} = 1/c_{min}$. The distances in real space Δx_1, Δx_2 and Δx_3 are the differences in the x, y and z coordinate of the two nodes, while the distances in phase space Δp_1, Δp_2 and Δp_3 are the differences in the $\frac{\partial T}{\partial x}$, $\frac{\partial T}{\partial y}$ and $\frac{\partial T}{\partial z}$ components of the slowness vector $\mathbf{p} = \nabla T$. Note that the unified distance σ is dimensionless. For the length l, the longest edge of the model which is perpendicular to the dominant propagation direction of the wavefront is chosen and c_{min} is the smallest velocity in the model. This means that the unified distance between two nodes at the same position ($\delta x = 0$) but furthest apart in the slowness space, (i.e. the space spanned by the components of the slowness vectors) is the same as between two nodes on the wavefront with the same slowness vector ($\delta p = 0$) but furthest apart from each other in real space. This is similar to the case for the wavefront tracking in two dimensions where the two metric coordinates x and y are scaled so that they also lie in $[-\pi, +\pi]$ (see section 3.1).

A reference value σ_{ref} for the unified distance in phase space is provided by the user and an edge split is performed if the distance given by (6.6) between two nodes is greater than $2\sigma_{ref}$. An edge contraction is performed if the distance between two nodes is smaller than $\sigma_{ref}/2$. As mentioned previously, the initial icosahedron is refined prior to wavefront propagation, so that for all the triangles, the distance along their edges in phase space is smaller than the reference value. Once this process is completed, the wavefront is propagated, and after each time step, it is determined whether or not triangles need to be added or removed.

In figure 6.10 a low velocity anomaly of up to 22 % perturbation has been superimposed on a background velocity distribution given by a linearly interpolated continuum between 3.0 km/s at the surface and 5.0 km/s at the lower boundary. The time step is 0.05 s and the reference distance σ_{ref} is 0.075. After 240 iterations, the wavefront has completely departed the computational domain. As one

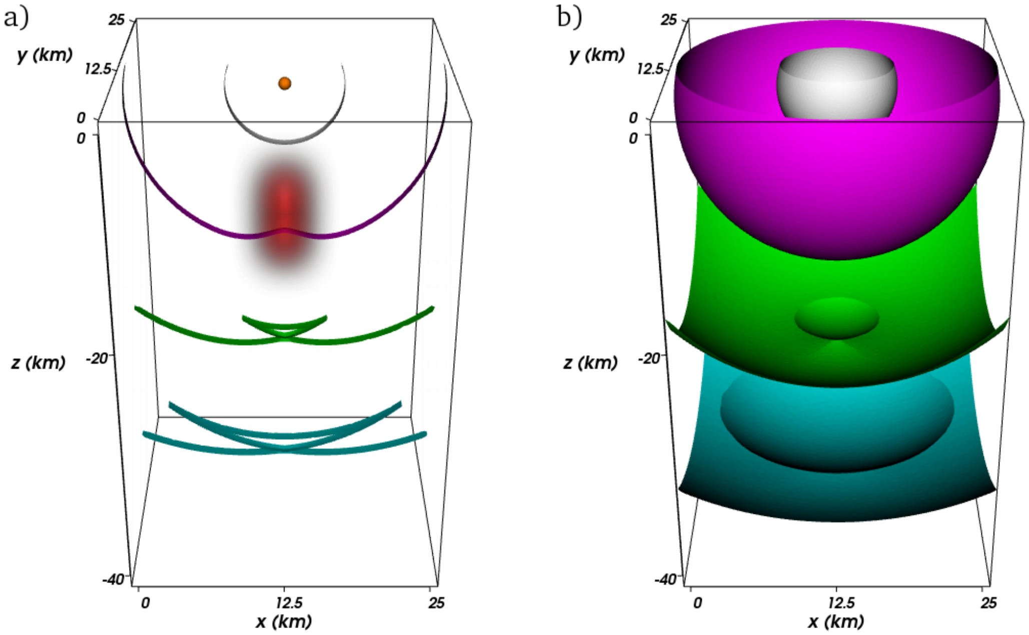


Figure 6.10: Low velocity anomaly of up to 22 % perturbation, which has been superimposed on a background velocity distribution, given by 3.0 km/s at the surface and 5.0 km/s at the bottom, and a constant velocity gradient in between; (a) shape of the low velocity anomaly and vertical slices through the wavefronts, (b) complete wavefronts plotted at 2.5 s intervals.

would expect based on the results from the two dimensional case (see chapter 3), the wavefront triplicates and the three dimensional equivalent of a swallowtail is generated.

Figure 6.11 shows the number of nodes as a function of iteration or time step for three different scenarios: (1) no surface refinement or simplification; (2) surface refinement but no simplification; and (3) surface refinement and simplification. For all three scenarios the number of nodes initially stays constant and then begins to decrease as parts of the wavefront begin leaving the computational domain. If no simplification or refinement is performed, the number of nodes decays monotonically as more and more nodes leave the computational domain. If surface refinement is used, the number of nodes begins to increase as the wavefront starts to triplicate (i.e. after 50 iterations of the time step). As the wavefront continues to propagate, the number of triangles eventually decreases as the nodes leave the computational domain and no new triplications are formed. However there is still a small increase between iterations 150 and 200 due to the further growth of the swallowtail. It is interesting to note that due to surface simplification, about 500

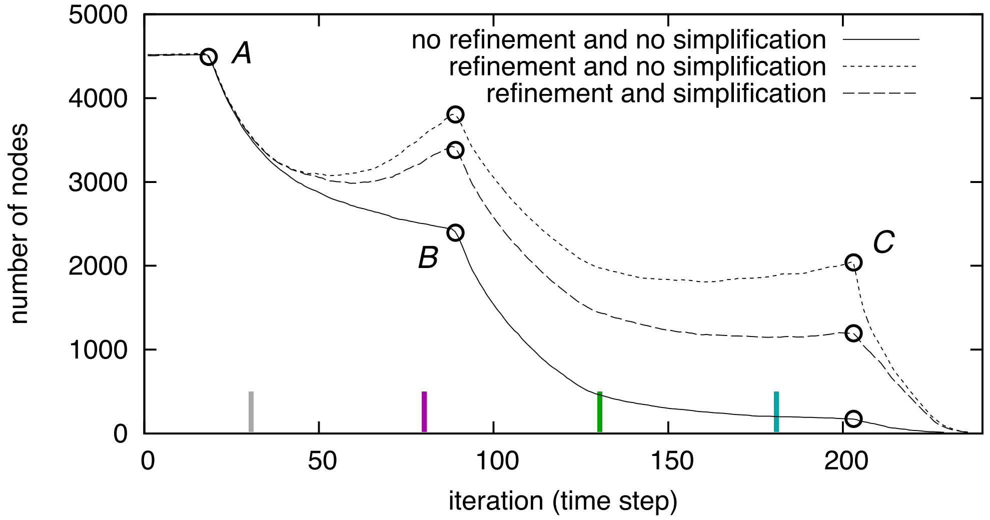


Figure 6.11: Number of nodes versus iteration step for the three different approaches to evolve a wavefront surface. The four iteration steps for which the wavefront in figure 6.10 is plotted, are marked by a line in the corresponding colour. The three points A, B and C where the curves have discontinuous gradients correspond to times when the wavefront begins to penetrate different parts of the computational domain and nodes begin to get removed. Refer to text for more detail.

nodes are removed from the wavefront once the triplication has been formed.

In all three scenarios in figure 6.11 three points can be located where the number of nodes has a discontinuous gradient. Each of these changes is associated with a different part of the wavefront reaching the boundary of the computational domain, and hence nodes begin to be removed from wavefront at a larger rate than they are added by surface refinement for parts of the wavefront within the computational domain. At point A, the top side of the initially spherical wavefront reaches the surface. The abrupt change in the number of points at point B is caused by the wavefront reaching the boundaries of the computational domain in the x and y direction. At point C the wavefront has reached the bottom of the computational domain. Figure 6.11 suggests that refinement is more dominant than simplification in that the number of nodes always increases and that a runaway is only prevented by the size of the computational domain. This behaviour is expected, because as the surface area of the wavefront grows, more and more nodes are needed to maintain a predefined density of points on the wavefront. Nevertheless, surface simplification is effective in reducing the number of nodes needed to maintain a fixed density of points on the wavefront, as figure 6.11 clearly illustrates.

The new method is implemented under GNU/Linux in Fortran and the follow-

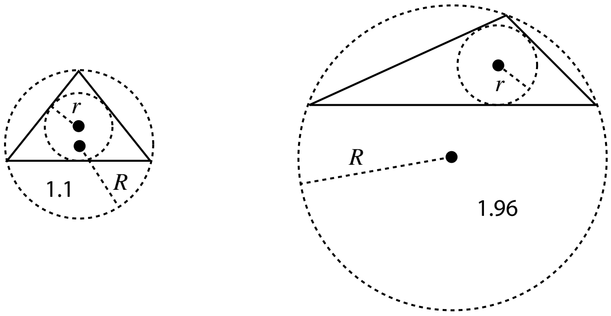


Figure 6.12: Two triangles and their radius ratio which is given by circum-radius R divided by twice the in-radius r of the triangle.

ing computation times are given for a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory. The computation time for the velocity model in figure 6.11 is 3.9 s if no refinement or simplification of the wavefront is carried out, 6.9 s if only refinement of the wavefront is performed, and 7.6 s if refinement and simplification is applied to the wavefront. The computation time appears to be largely unaffected by the use of simplification (in fact, it increases slightly). This is due to the fact that verifying whether a triangle can be removed after every time step requires additional calculations. It turns out to be more efficient to perform a surface simplification every m-th time step; for example a computation time of 4.4 s is achieved for $m = 25$. This configuration was used to obtain the wavefronts plotted in figure 6.10. The optimum strategy for when to refine and simplify a surface is ultimately dependant on how the surface evolves. Testing all the triangles after each time step might require more computation time than carrying along a few unnecessary triangles for several time steps before removing them (as demonstrated above).

A potential disadvantage of the modified butterfly scheme used here is that for complex wavefronts, one might end up with a mesh containing many ill-shaped or distorted triangles, because no global optimisation of the wavefront triangulation is performed. The quality of the triangulation is investigated by using the radius ratio $R/(2r)$, where R is is the circum-radius and r the in-radius of a triangle (see figure 6.12). For an optimum triangulation, each triangle is equilateral and hence would have a radius ratio of 1.0 (e.g. Pébay & Baker, 2003). The larger the radius ratio the more ill-shaped the triangle.

Figure 6.13a shows a view from the top onto the wavefront in figure 6.10 after 9 s. The new nodes have been inserted using linear interpolation and the triangles are coloured according to their radius ratio. The triangulation for the portion

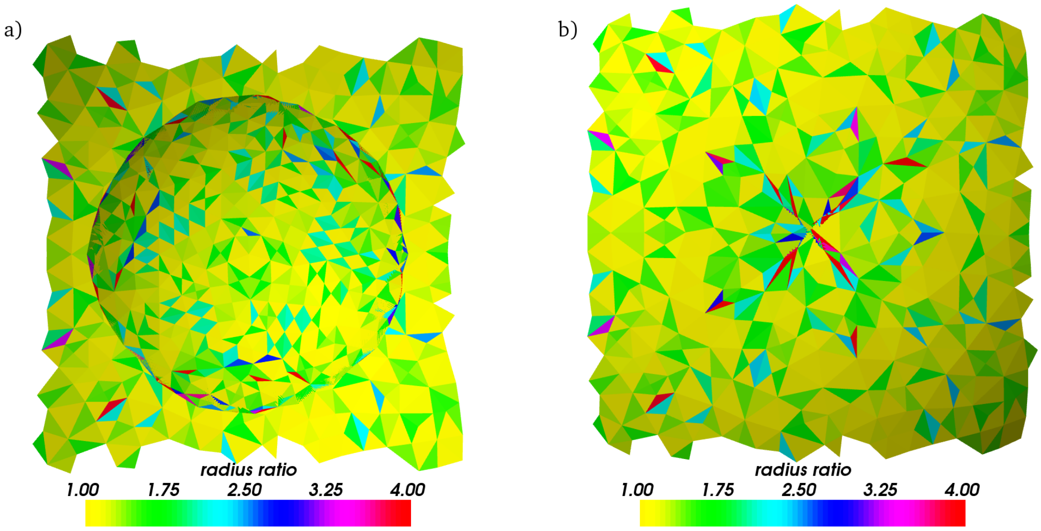


Figure 6.13: The wavefront (see figure 6.10) seen from the top (a) and bottom (b). Linear interpolation has been used to add new nodes to the wavefront. In this and the following four figures the wavefront is plotted after 9 s and the triangles are coloured according to their radius ratio.

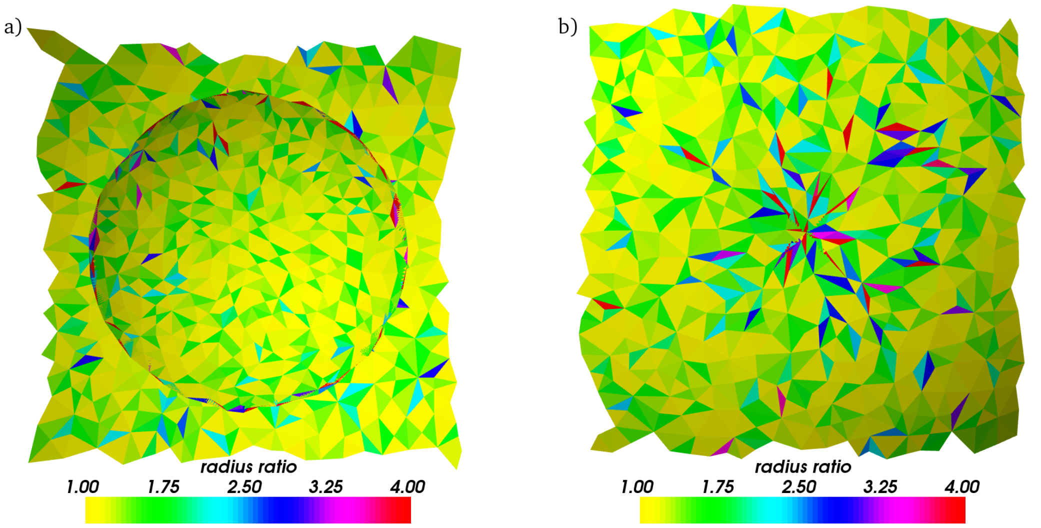


Figure 6.14: Same as figure 6.13 except that the modified butterfly scheme has been used to add new nodes to the wavefront.

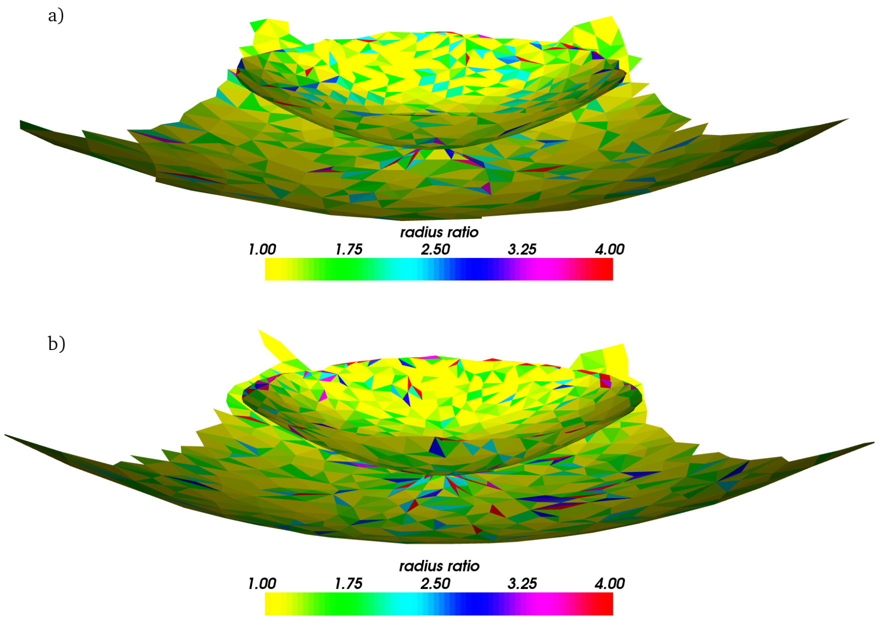


Figure 6.15: View of the wavefront (see figure 6.10) from the side. (a) Result from linear interpolation; (b) result from the modified butterfly scheme

of the wavefront responsible for the third arrival is very well behaved. Only at the sharp edge of the wavefront (i.e. the endpoints of the swallowtail) is it possible to observe triangles with a large radius ratio. Overall, ill-shaped triangles seem to be a rare occurrence. Figure 6.13b shows a view of the same wavefront from below, where several ill-shaped triangles are present. These are concentrated near the centre, where the wavefront is self-intersecting i.e. at the caustic point. When the new nodes are added to the triangulation using the modified butterfly scheme (figure 6.14), the triangulation in the top and bottom view of the wavefront is not significantly improved. When looking up to the wavefront from the bottom (figure 6.14b), there are more triangles with larger radius ratio compared to figure 6.13b.

Figure 6.15 shows a side view of the wavefront at 9 s. Where the wavefront self-intersects, the modified butterfly scheme leads to a smoother surface. From this view point it also looks as though the modified butterfly scheme leads to a smoother representation of the circular edge associated with the swallowtail.

In figures 6.16 and 6.17 the wavefronts are plotted in slowness space i.e. using the three components of the slowness vector as the axis of the coordinate system. The slowness vector only changes gradually across the surface, with the exception of nodes near the circular shaped edge of the swallowtail. Therefore, one expects

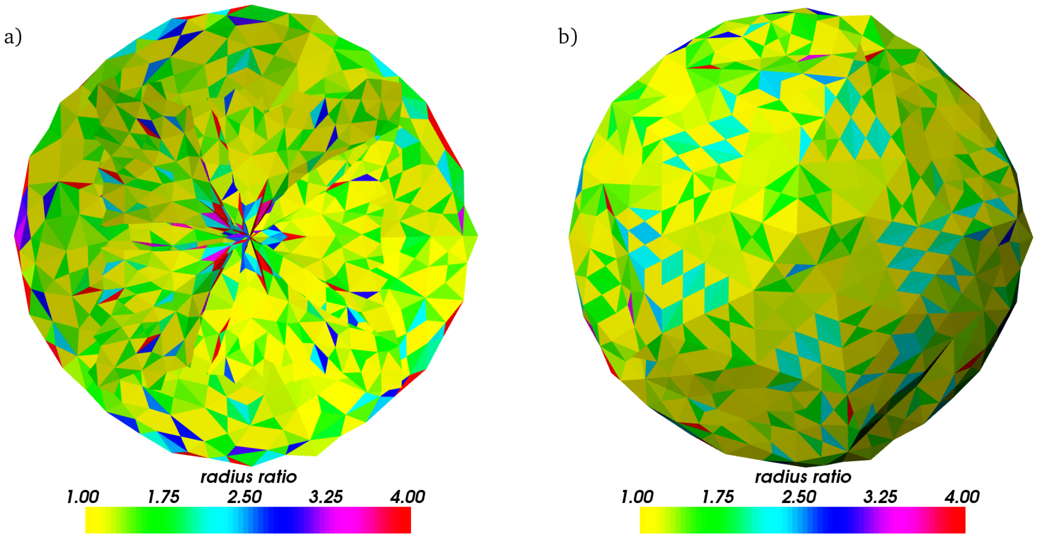


Figure 6.16: Wavefront in slowness space seen from the top (a) and bottom (b). Linear interpolation has been used to add new nodes to the wavefront.

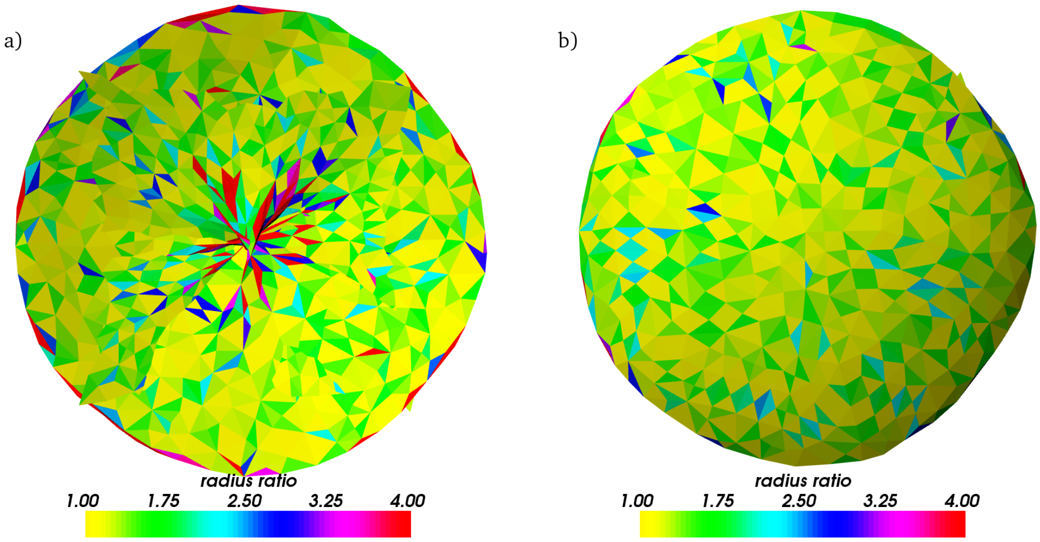


Figure 6.17: Wavefront in slowness space seen from the top (a) and bottom (b). The modified butterfly scheme has been used to add new nodes to the wavefront.

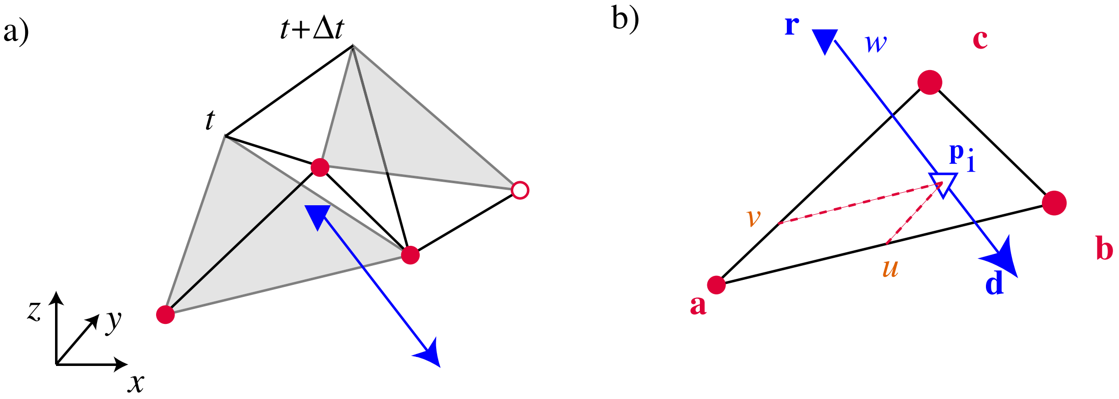


Figure 6.18: (a) Prism shaped polyhedron formed between the triangles on the two consecutive wavefronts for the times t and $t + \Delta t$. The number of faces crossed by tracing along a line starting at the receiver (blue triangle) can be counted. If the resulting number is odd the receiver is inside the polyhedron. (b) To determine whether or not the line intersects a given face, its intersection with the plane given by the triangle is found. If the intersection point is inside the triangle, then the line intersects the face.

to see a smooth disk like structure. The modified butterfly scheme leads to an overall smoother surface. If linear interpolation is used, several edges appear in the surface which are not present when the modified butterfly scheme has been used to add new nodes to the wavefront. The number of triangles with larger radius ratios seems to be independent of the interpolation scheme used.

In summary, the modified butterfly scheme leads to a small improvement in the triangulation, and especially in slowness space, the resulting surface seems to be smoother. However, as far as mesh quality is concerned, linear interpolation is probably sufficient for most practical applications.

6.1.4 Extracting arrival information

For two consecutive wavefronts at times t and $t + \Delta t$, and the ray path segments between the edges of triangles on the wavefront, a set of prism shaped polyhedra can be defined. In order to extract source-receiver arrival times, the polyhedron in which a receiver is located needs to be identified. This essentially amounts to testing if a point lies inside a polyhedron. The so called crossing method, discussed earlier to solve the equivalent problem in two dimensions (section 3.1.2) can be extended to three dimensions. As before, a line is traced and the number of faces it intersects counted. If the number is even the point lies outside the polyhedron, and if the number is odd the point lies inside the polyhedron.

Figure 6.18a illustrates this approach. To determine whether or not the line intersects a triangular face of the polyhedron, the intersection point of the line with the plane of the triangle is calculated. If the intersection point lies inside the triangle, the line intersects the face of the polygon. The line-triangle intersection point is computed using parametric coordinates for the intersection point in the plane. The intersection point $\mathbf{p_i}$ is given by

$$\mathbf{p}_i = \mathbf{a} + u(\mathbf{b} - \mathbf{a}) + v(\mathbf{c} - \mathbf{a}) = r + w\mathbf{d}, \tag{6.7}$$

where $\mathbf{a}$, $\mathbf{b}$ and $\mathbf{c}$ are the position vectors of the three corners of the triangle, $\mathbf{d}$ the direction of the line starting at the receiver $\mathbf{r}$ and u,v and w are the parametric coordinates. In three dimensions (6.7) forms a linear system of equations with three unknowns and three equations. If the intersection point lies inside the triangle the line intersects the face of the polyhedron; if it is outside the line does not intersect the face of the polyhedron.

Three of the five faces of the prism shaped polyhedron are defined by four nodes. As these four points do not necessarily lie in the same plane, these faces are split into two triangles (as shown in figure 6.18). The shortest of the two diagonals is used to split the square into two triangles. This guarantees that when the neighbouring prism is tested, the same triangulation of the shared face is used. It is also important to use the wavefront at time $t + \Delta t$ before surface refinement and simplification are applied, as otherwise there would no longer be the same set of triangles on both wavefronts.

Once it is determined that a receiver lies inside a polyhedron, an arrival time and slowness vector can be interpolated at the receiver (figure 6.19). The time and position of the two wavefront segments are known, so the shortest distances from the receiver to each wavefront d_a and d_b can be calculated. Using these two distances, an arrival time t_r can be linearly interpolated as $t_r = t + \Delta t d_a/(d_a + d_b)$. This method used for the extraction of arrival times is similar to the approach adopted by Vinje et al. (1996a).

To interpolate a slowness vector for the arrival, a slowness vector for the points c_a and c_b is first computed (see figure 6.19). The points lie on the same plane as the two triangles and are closest to the receiver. These two points can be located using (6.7), where $\mathbf{d}$ is given by the normal to the triangle. The position of the points c_a and c_b can also be expressed using the parametric coordinates u and v, defined along the edges of the triangle. Assuming that the gradient of the slowness

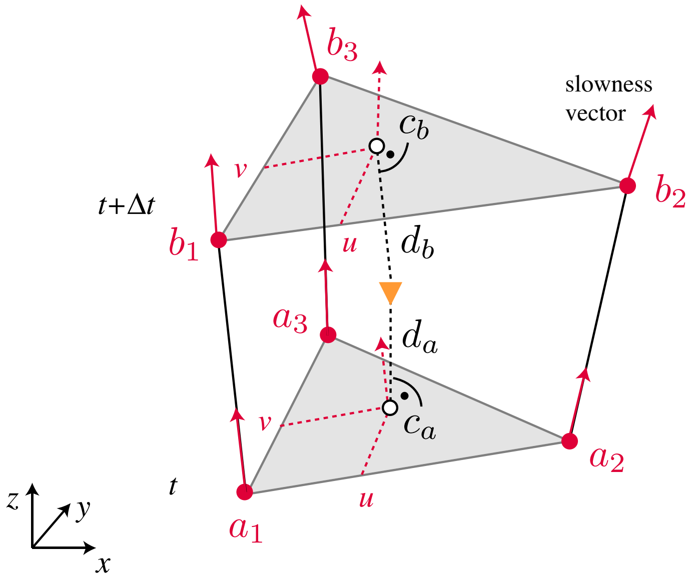


Figure 6.19: Illustration of the interpolation method used to compute an arrival time and corresponding slowness vector at a receiver (orange triangle) located between two consecutive wavefronts.

vector is constant within the area spanned by a triangle, a slowness vector at c_a can be computed using linear interpolation. Similarly, one can interpolate a slowness vector at c_b. Once the slowness vectors for c_a and c_b are calculated a slowness vector at the receiver using the shortest distances d_a and d_b is approximated in the same way as the travel time. The additional step of calculating a slowness vector at c_a and c_b is needed because wavefronts are only iso-contours of the travel time and not the slowness vector.

In two dimensions, the wavefronts were stored for different times and used to back track a ray from the receiver to the source (section 3.1.2). In three dimensions, the back tracking could be done using the triangles on the wavefront. Implementing such a scheme would be quite difficult considering that triangles are continuously added and removed from the wavefront during its propagation. The wavefront for every time step would also need to be stored on disk. An alternative is to actually shoot a ray from the receiver to the source. The initial direction of the ray at the receiver is given by a vector pointing in the opposite direction to the slowness vector. The only drawback is that the resulting ray may not intersect the source, as the initial ray direction may not be accurate enough due to using linear interpolation at the receiver and accumulation of error in the slowness vector as the wavefront propagates.

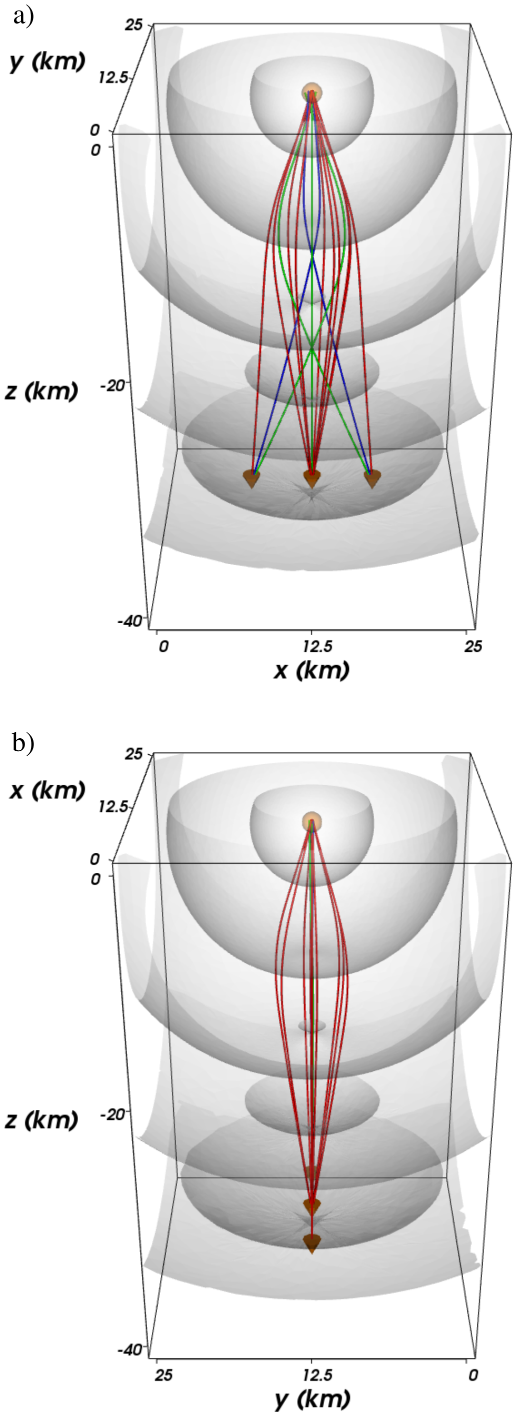


Figure 6.20: Ray paths corresponding to first, second and third arrivals (plotted as red, green and blue lines, respectively) for the wavefront model shown in figure 6.10. For the receiver in the centre, only a sample of five possible first arrival ray paths has been plotted. The first wavefront is plotted after 1.5 s and the following ones are plotted at 2 s intervals. The source is given by the orange sphere and the receivers by the dark orange cones. (a) View along the y-axis, (b) view along the x-axis. The velocity structure is shown in figure 6.10a.

Figure 6.20 shows ray paths computed using the technique of tracing rays back to the source, as outlined above, for the low velocity anomaly example (figure 6.10) for three receivers. Due to the symmetry of the problem, only two arrivals are observed at the receiver immediately below the source. The first arrival is observed when the point where the wavefront self- intersects passes through the receiver. In two dimensions, there are only two ray paths for this arrival. In three dimensions, however, there are an infinite number of ray paths (a sample of five is shown in figure 6.20). They form a surface defined by rotating one of the ray paths for the first arrival around the straight line between source and receiver. The second arrival is given by the back-end of the swallowtail hitting the receiver. Note that not all the ray paths end exactly at the source position. This is because the ray is traced back to the source and not extracted from the wavefronts. If more robust ray paths are needed, one would have to calculate them by back tracking a ray using the previously computed wavefronts.

6.2 Examples

First, the accuracy of the new wavefront tracking approach is measured for a model with a constant velocity gradient. The mean error and the computation time are mapped as a function of the size of the time step and the reference distance. Afterwards, the capability of the method to compute wavefronts in complex velocity models is investigated using a random velocity field and a smoothed version of the SEG/EAGE salt dome model. The scheme has been implemented under GNU/Linux in Fortran and all computation times are given for a Pentium 4 CPU running at 3.2 Ghz with 3 Gb of memory.

6.2.1 Constant velocity gradient

In section 3.2.2 the accuracy of the Lagrangian solver was discussed using a structure with a constant velocity gradient, where travel times can be computed analytically. Figure 6.21 shows the ray paths for a three dimensional model given by a constant vertical velocity gradient. The velocity at the top is 4.75 km/s and 7.75 km/s at the base. For this velocity structure, the area of each triangle grows monotonically as the wavefront expands. Therefore, surface simplification will not remove any triangles.

A grid of receivers with a uniform spacing of 2 km is placed at the surface and

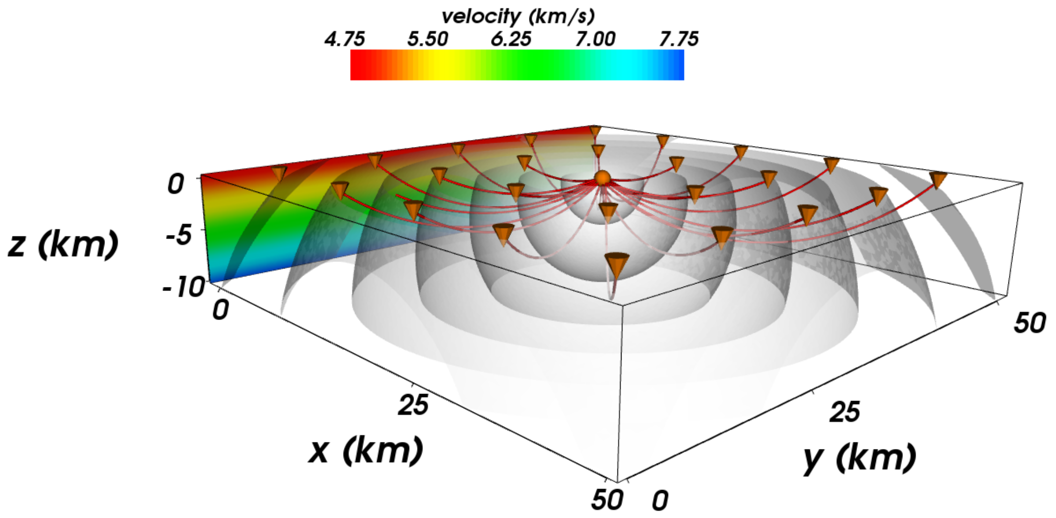


Figure 6.21: Ray paths for a model in which the velocity increases linearly with depth. The source is marked by a sphere and cones denote the receivers (although not all are shown).

the computed travel times compared with the analytical solutions. There are no receivers within a 5 km by 5 km square around the source. In figures 6.22 and 6.23, the average relative error for the 324 receivers is mapped as a function of the two independent parameters, the size of time step and the reference distance in phase space (section 6.1.3). Note that as the reference distance decreases, the number of nodes used to describe the wavefront increases.

The accuracy of the travel times depends on the time step, and on the number of nodes and triangles on the wavefront to approximately the same degree. In the two dimensional case (section 3.2.2) the relative errors are smaller and depend less on the size of the time step. Computation time is influenced by both the time step and the reference distance. For larger reference distances, the computation time depends more on the time step. This is similar to the situation in two dimension, where the time step is the dominant factor (cf. section 3.2.2).

In order to keep a fixed density of points on the wavefront, each edge of every triangle needs to be tested separately. Since each node is likely to belong to more than one triangle, greater computation time is required to maintain a fixed density of points on the wavefront than to update the positions of these points for a given time step. In two dimensions, measuring the distance between two points on the bicharacteristic strip requires a smaller amount of computation time than updating the position of the two points. This explains why in two dimensions the computation time is largely a function of time step. The computation times are also

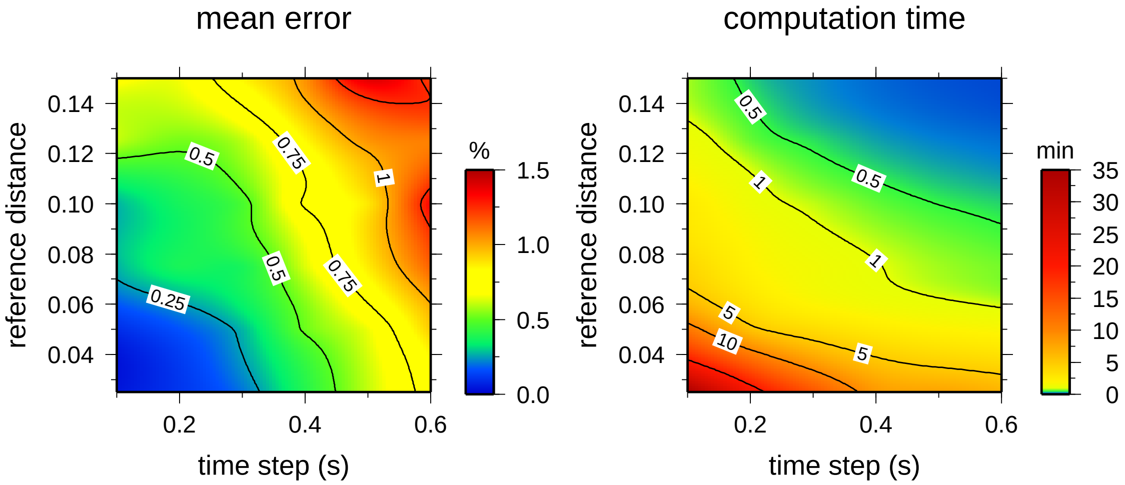


Figure 6.22: Average difference between the analytical and numerical solution (left) and computation time (right) for different time steps and dimensionless reference distances. New nodes have been added using linear interpolation.

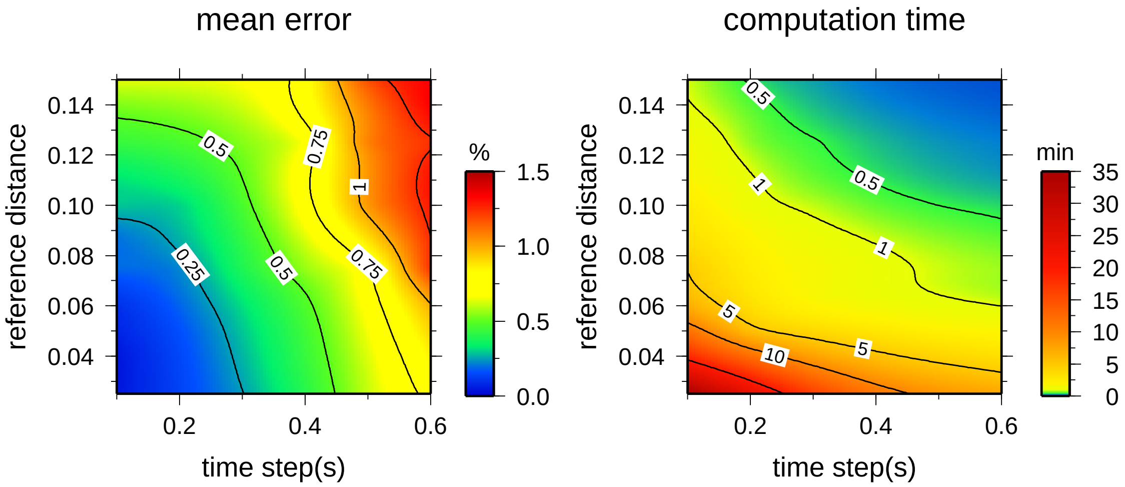


Figure 6.23: Average difference between the analytical and numerical solution (left) and computation time (right) for different time steps and dimensionless reference distances. New nodes haven been added using the modified butterfly scheme.

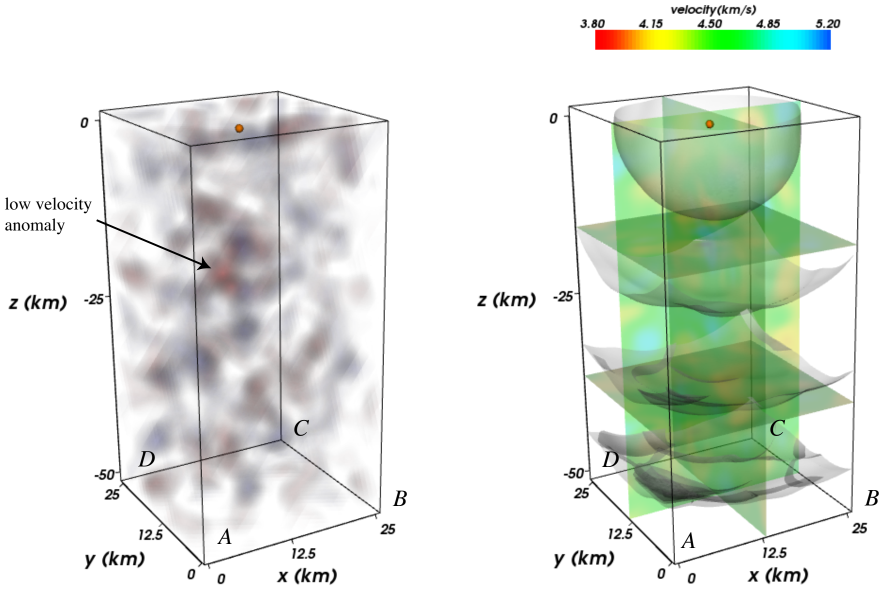
larger than those reported for the low velocity anomaly (previous section), where only the wavefronts were tracked. This is because having to verify whether the 324 receivers are located between the corresponding triangles of two consecutive wavefronts requires significant computational effort.

In figure 6.22 linear interpolation scheme is used for adding nodes while in figure 6.23 the modified butterfly scheme is used. The modified butterfly scheme leads to a small decrease in the mean error but also to a small increase in computation time, especially for larger numbers of nodes and triangles (i.e. a small dimensionless reference distance). Considering that it also leads to a slightly better triangulation, the case for using a higher order interpolation scheme might be stronger for wavefront tracking in three dimensions than in two dimensions, particularly when the medium is complex. However, the overall accuracy achieved by linear interpolation is not significantly lower and the dominant source of error is likely to be the linear interpolation of an arrival time at a receiver.

6.2.2 Complex structure

A three dimensional velocity model based on random variations in wave speed was generated in order to test the ability of the scheme to handle complex wavefronts, featuring several triplications that intersect each other. The velocity model has dimensions of 25 × 25 × 50 km and is described by a uniform grid of velocity nodes with a spacing of 2.5 km (see figure 6.24). The background velocity is set at 4.5 km/s. At a few selected nodes the velocity has been perturbed by up to 50 % (i.e. 2.25 km/s). The nodes are selected so that triplications starts to develop early during wavefront propagation. Note that there is a low velocity zone above the centre of the model which causes the triplication located in the vicinity of the DB diagonal (figure 6.24). In addition, Gaussian random noise with a standard deviation of 0.45 km/s is superimposed onto the model to cause the development of additional swallowtails. The final velocity model is shown in figure 6.24.

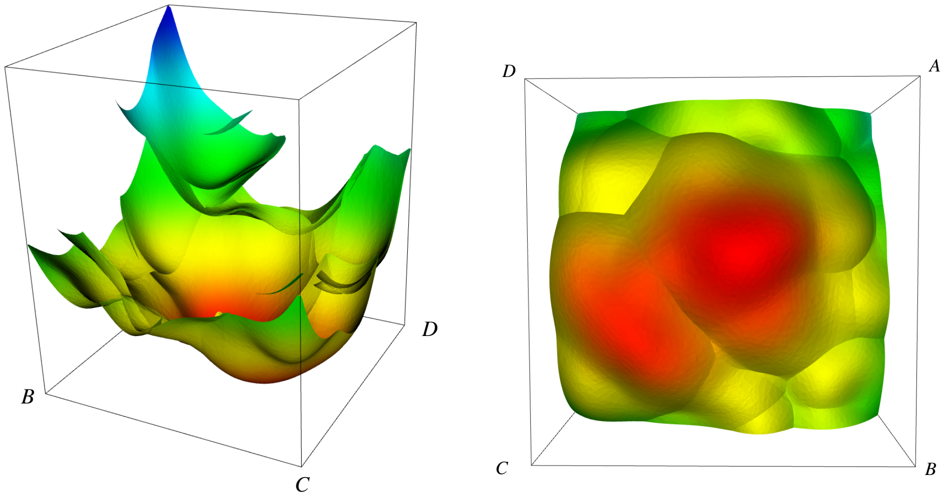
The basic wavefront evolution parameters are as follows: the time step is 0.05 s, the dimensionless reference distance in phase space is 0.05, and 230 iterations are performed. The computation time is 56 s if only the wavefront is tracked and no receivers are present. The velocity field causes the wavefront to triplicate several times (see figure 6.24b). Figure 6.25 shows the wavefront at 10 s; as expected several swallowtails overlie each other. When viewed from the bottom of the model all visible portions of the wavefront correspond to the first arrival surface



(a)

(b)

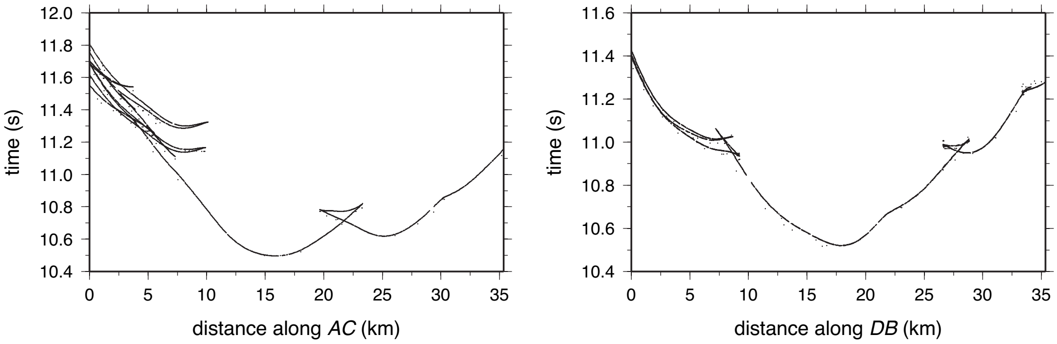
Figure 6.24: Random velocity model with a distinct low velocity region and a source near the surface (orange sphere). (a) Regions of the model with a velocity below 4.5 km/s are shaded red, while regions with a velocity above 4.5 km/s are shaded blue. (b) Slices through the velocity model and wavefronts. The time interval between two consecutive wavefronts is 2.5 s.



(a)

(b)

Figure 6.25: Snapshot of the wavefront after 10 s seen from two different angles (a) from the top and (b) from the bottom. The wavefronts are coloured according to height and plotted with a vertical exaggeration of 4.



(a) arrival times along AC (b) arrival times along DB

Figure 6.26: Arrival times as observed along the two diagonals at the bottom of the cube (see figure 6.24).

(figure 6.25b). This surface is characterised by a distinct pattern of V-shaped valleys. These valleys are due to wavefront self-intersections that are linked with the development of swallowtails caused by the velocity field. The valleys represent caustic lines where rays intersect each other (see section 3.1.3).

Travel time information associated with the propagating wavefront is examined by placing 1250 receivers along each of the two diagonal lines AC and BD along the bottom of the cube (figure 6.24). Verifying for each time step if one of the triangles on the wavefront crosses a receiver leads to an increase in computation time. Even if arrivals are searched for within an a priori time span (i.e. between 10 and 12 s), the computation time is still 105 min, due to more than 50, 000 nodes being used to describe the wavefront. Figure 6.26 shows the corresponding arrival time curves. Overall the arrival time curves reflect the complex geometry of the wavefront. Several swallowtails can be observed and for the receivers near the corner A, up to seven arrivals are successfully tracked. However the travel times curves are afflicted by several gaps and incorrect travel times. These problems are likely to be caused by numerical instabilities associated with the point in polyhedron part of the method, which need to be addressed if real data applications are to be considered. Nevertheless, this example demonstrates that the scheme can be used to compute multi valued travel times for three dimensional structures.

6.2.3 SEG/EAGE Salt dome model

The three dimensional SEG/EAGE salt dome velocity model (e.g. Aminzadeh et al., 1996) is based on features that are characteristic of complex salt structures in the Gulf of Mexico. The model is dominated by a plunging salt stock (see figure 6.27). The dominant features of the salt stock are a rounded overhang and a faulted flank opposite the plunging part of the structure. The sediments are dominated by thin sand layers and sand lenses with different velocities. The wave speed contrast between the salt and the surrounding sediments is more than 30 % (i.e. 1000 m/s). The model covers an area of 13.5 × 13.5 × 4.18 km and the grid spacing for the velocity values is 20 m. This structure has previously been used to benchmark new techniques for the computation of first arrival travel times (e.g. Alkhalifah & Fomel, 2001; Shin et al., 2003).

If the wavefront tracking scheme is applied to the original version of the SEG/EAGE Salt dome model, the wavefronts become extremely complex due to significant changes in velocity from one grid node to the next. Consequently, memory becomes an issue with the number of triangles and nodes simply too large to fit into 3 Gb. Therefore, as is commonly done when benchmarking new ray or wavefront propagation techniques, the velocity model is smoothed (e.g. Buske & Kästner, 2004).

Bulant (2002) discusses different approaches to optimally smooth velocity models for ray tracing and in particular focuses on the SEG/EAGE salt dome model. The software provided by the consortium SW3D (see `http://sw3d.mff.cuni.cz`), was used to sample the salt dome model with a grid spacing of 200 m using an average velocity for each cell. The smooth version of the model used here is slightly smaller than the original version as it covers a volume of 13.4 × 13.4 × 4.0 km. This process is likely to remove all small scale features of the model and will lead to simpler wavefronts. However, the complex salt sediment interface that remains generates significant multipathing (i.e. several later arrivals). The smooth version of the velocity model and the source location is shown in figure 6.27.

Figure 6.28 and 6.29 show different views of the wavefronts at two different moments in time for a source located at $(6740, 6740, -250)$. The time step is 0.0025 s and the reference distance in phase space 0.025. The computation time for the blue wavefront in figure 6.28 is 13 min.

As soon as parts of the wavefront reach the salt, the wavefront begins to triplicate and develops one major swallowtail pattern, which is circular shaped around the source and only interrupted where the overhang of the salt dome is located (figure 6.28 and 6.29). The wavefront eventually becomes very complex with several smaller swallowtails superimposed on the major swallowtail. The swallowtails also start to intersect each other. Despite having smoothed the model significantly the resulting wavefronts are still complex.

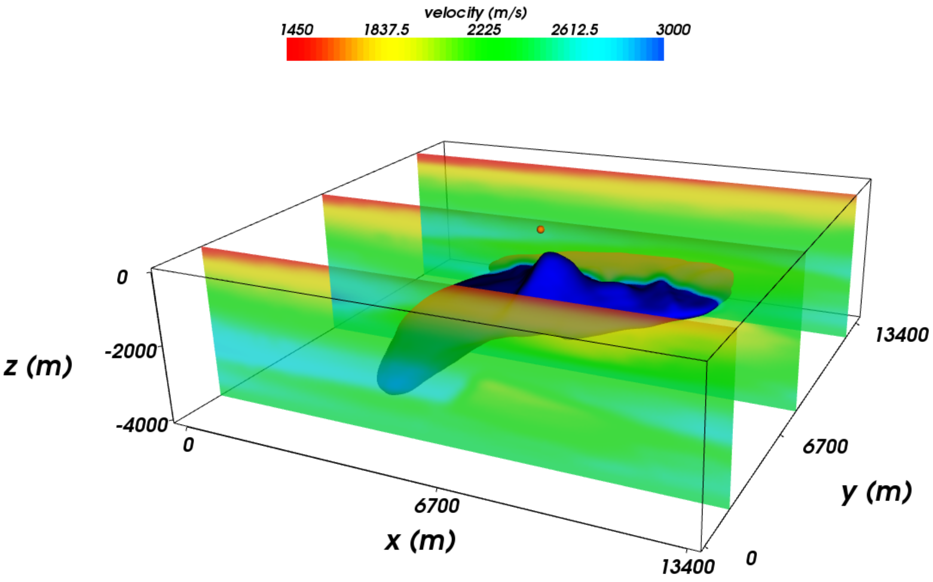


Figure 6.27: Slices through the smoothed version of the SEG/EAGE salt dome model. The 3000 m/s iso-surface corresponds roughly to the boundary of the salt dome.

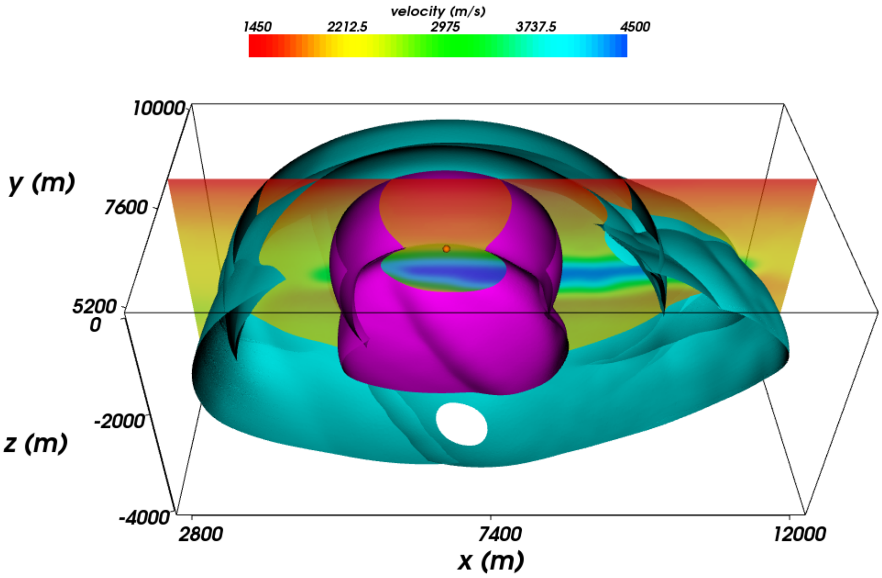


Figure 6.28: Two snapshots of the wavefront computed in the SEG/EAGE salt dome model. The orange sphere denotes the source location. The wavefront is plotted at 3.375 s (magenta) and 4.125 s (cyan). Part of the wavefront surface has been removed to facilitate visualisation.

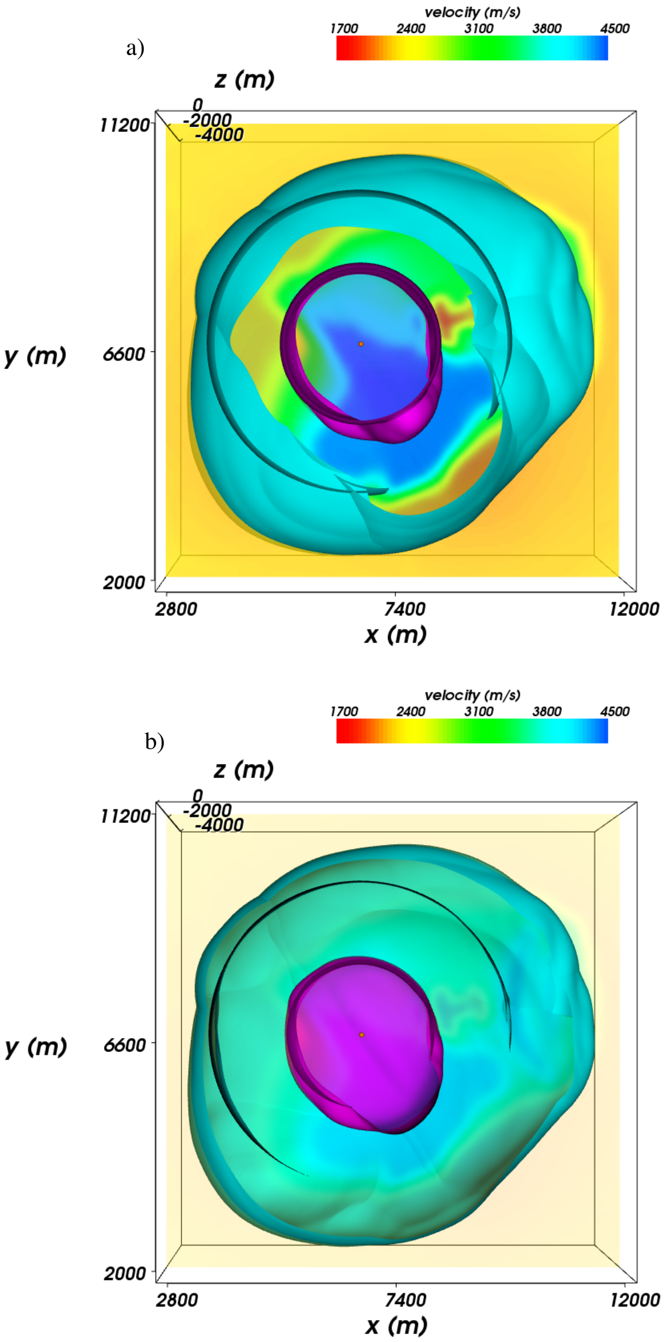


Figure 6.29: Two snapshots of the wavefront computed in the SEG/EAGE salt dome model. The orange sphere denotes the source location. The wavefront is plotted at 3.375 s (magenta) and 4.125 s (cyan). The slice though the velocity model is at a depth of −1500 m. (a) Top view of the evolving wavefront. (b) Same view as (a) but the top part of the two wavefronts has been removed and the slice through the velocity model is more transparent.

The results for the SEG/EAGE salt dome model show that the mesh refinement and simplification strategy chosen in this work can handle wavefronts which become far more complex than those computed with previous wavefront techniques (e.g. Vinje et al., 1999; Lucio et al., 1996). It remains to be seen whether later arrivals could be used to improve seismic imaging of such complex structures.

6.3 Summary

In this chapter we have demonstrated that an extension of the Lagrangian approach to three dimensions is entirely feasible. The suggested refinement and simplification schemes adopted from sophisticated computer graphics surface evolution techniques lead to smooth and well behaved wavefronts even for strongly heterogeneous models. Two complex models have been used to demonstrate the potential of the new approach for wavefront tracking in three dimensions. The scheme can track multi valued wavefronts through complex structures in a reasonable amount of computation time. The wavefront stays smooth and well behaved even in the presence of strong velocity gradients, as demonstrated for the SEG/EAGE salt dome model. Ultimately the complexity and resolution of the wavefronts is only limited by the availability of memory and computation time. A model with a constant velocity gradient was used to demonstrate the accuracy of the new scheme. When compared to the two dimensional case the accuracy is lower but of the same order of magnitude.

Using a point-in-polyhedron approach for the extraction of arrival times at a receiver increases the computation time significantly, as one has to verify whether each receiver lies between two consecutive triangles of the wavefront after each time step. On the other hand, wavefronts computed during wavefront propagation are isochrons of the travel time field and known for discrete time steps. A more efficient approach for the extraction of travel times could be to interpolate a multi valued travel time field once the wavefront propagation is finished.

For many applications, the accuracy of the ray paths computed by ray tracing from the receiver to the source, based only on an initial direction at the receiver given by an interpolated slowness vector, will be sufficient. More accurate ray paths could always be obtained by tracking the rays back from the receiver using the wavefronts computed during the propagation process. Once ray paths are extracted, the scheme could be used for seismic tomography with later arrivals in three dimensions.

Instead of just tracking the wavefront from a point source, the scheme could easily be adapted for an incoming plane wave. The only requirement would be to triangulate the initial plane wave, for example by using Delaunay triangulation (e.g. Okabe et al., 1992). Although in theory it would be feasible to develop the three dimensional Lagrangian scheme to same level of sophistication as its two dimensional counterpart (see chapters 3 and 4) by including interfaces and calculating Gaussian beams, significant effort beyond the scope of this thesis would be required.

Chapter 7

Conclusions

A seismic wavefront that travels through a heterogeneous velocity medium may develop a swallowtail, which arises when the wavefront distorts to such an extent that it self-intersects. If only first arrivals are required then a variety of efficient solvers which discard later arriving information, such as finite difference solution of the eikonal equation and shortest path ray tracing, can be applied. The challenge of computing later arrivals lies in successfully tracking, either implicitly or explicitly, a complex self-intersecting surface that may also include sharp corners. A novel approach to mitigating these difficulties, developed in this thesis, is to unfold the wavefront into reduced phase space or full phase space. This has the benefit that the bicharacteristic strip (i.e. the wavefront in phase space or reduced phase space) is single valued and locally smooth, even for a self-intersecting wavefront with sharp corners.

In chapter 2 an Eulerian approach for the computation of multi valued travel times is implemented. The decision to explore the potential of a grid based scheme to calculate multi valued travel times was based on the demonstrated advantages of the fast marching method compared to conventional ray tracing (e.g. Rawlinson & Sambridge, 2004a; de Kool et al., 2006). The fast marching method is a grid based eikonal solver, in which the interface or wavefront is not permitted to cross a cell of the underlying grid more than once. Therefore it can be used for the calculation of first arrival travel times but not multi valued travel times. However, it would be beneficial to use the underlying principles of the scheme in the computation of multi valued travel times if possible.

The scheme proposed by Osher et al. (2002) attempts to do this via an implementation of the level set method. A detailed discussion of the level set method, its relationship to fast marching and the numerical schemes for solving the level set equation is given in section 2.2. The wavefront in this case is implicitly tracked in reduced phase space in order to compute multi valued travel times. It turns out that an efficient implementation of this technique is a problem in itself. In order to track a wavefront in two dimensions, two surfaces have to be evolved in three dimensions using the level set method. This requires that two initial value partial differential equations are solved on two three dimensional grids; consequently halving the grid size will increase computation time by a factor 16. More efficient implementations based on adaptive gridding and narrow bands (e.g. Losasso et al., 2006; Frolkovič & Mikula, 2007) are still under development. While they may eventually prove to be useful in practical seismic problems, they are inherently more complex than the explicit Lagrangian scheme presented in chapter 3. Moreover, it is difficult to envisage how even the most sophisticated finite difference solvers will achieve parity with the Lagrangian scheme in terms of accuracy versus computation time, given the need to solve two multi dimensional initial value partial differential equations. Of course the method proposed by Osher et al. (2002) is by no means the only grid based technique for multi valued travel times under development (e.g. Fomel & Sethian, 2002; Engquist & Runborg, 2003; Qian & Leung, 2004). The considerable impetus in the applied mathematics/computational physics community to advance this area of research may soon yield techniques suitable for practical application. However, for the moment at least, Lagrangian wavefront tracking as introduced in section 3.1 appears to be a superior approach.

Although it has been shown that the Lagrangian wavefront construction scheme is preferable in almost every respect to Eulerian techniques for tracking multipathing wavefronts, the level set method is a well established tool in its own right and superior to Lagrangian schemes in a wide variety of other interface tracking applications. For example it is widely used for simulating the evolution of interfaces between fluids or modelling rising gas bubbles in fluids (e.g. Chang et al., 1996; Losasso et al., 2006). One of the advantages a level set method has for these types of applications is that it handles the merging and breaking of interfaces automatically, compared to schemes where interfaces are represented explicitly. In the field of earth sciences the level set method is not an established technique, although Hale et al. (2007) use it successfully to model the endogenous growth of lava domes in an axisymmetrical formulation in cylindrical coordinates (i.e. a two dimensional problem is solved). Its shortcomings in seismic wavefront tracking can be attributed to its inability to directly represent a self-intersecting wavefront.

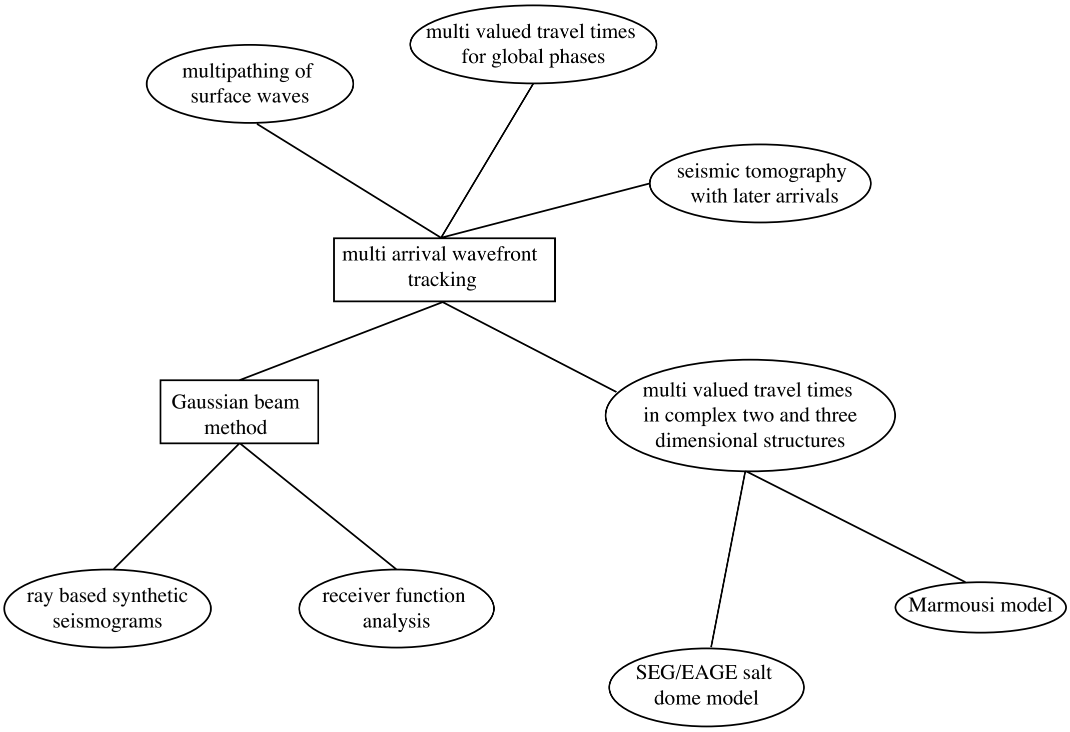

multi valued travel times for global phases
multipathing of surface waves
seismic tomography with later arrivals
multi arrival wavefront tracking
Gaussian beam method
multi valued travel times in complex two and three dimensional structures
ray based synthetic seismograms
receiver function analysis
Marmousi model
SEG/EAGE salt dome model

Figure 7.1: Overview of the different applications of the Lagrangian scheme for multi arrival wavefront tracking presented in this work.

The Lagrangian scheme developed in this work for wavefront tracking in two and three dimensions is robust, efficient and can be used for a wide variety of applications as demonstrated in the numerical tests carried out in chapters 3-6. Figure 7.1 presents an overview of the different seismic applications of multi valued travel times investigated in this work. In general, it can be used as an alternative to first arrival schemes for the computation of travel times in heterogeneous two and three dimensional velocity models, with the added bonus that later arrivals are included. Examples involving the Marmousi model and SEG/EAGE salt dome model show that the method remains stable even in the presence of velocity contrasts as large as 3:1, which produce over 60 distinct later arrivals.

Lagrangian wavefront tracking techniques in two dimensions have been developed previously (e.g. Lambaré et al., 1992; Vinje et al., 1993). The scheme presented here is different in that the wavefront is unfolded into reduced phase space, where it stays locally smooth. When it comes to interpolating new nodes, this smoothness is a distinct advantage, and occurs because the distance between two nodes in real space and the angle between the corresponding wavefront normals is combined into one unified reduced phase space distance. Nodes that define the wavefront can be both added or removed depending on whether their local density falls below or rises above predefined thresholds, respectively (see chapter 3).

The ability to remove points dynamically during the propagation process leads to greater efficiency, and further distinguishes the new scheme from previous work in the field. In addition to improving the basic wavefront tracking technique, the research carried out in this thesis also focuses for the first time on applications in solid earth seismology. Previously the focus has been on applications in the field of exploration geophysics (e.g. Vinje et al., 1993; Lambaré et al., 1996; Vinje et al., 1999). The migration of reflectors is one area of research where multi valued travel times computed by wavefront tracking has successfully been applied (Xu & Lambaré, 2004). Only travel times are required in this case, so the extraction of ray paths is generally not seen as a priority. The suite of examples presented in chapters 3-6 show that wavefront tracking can be a valuable tool at local, regional and global scales. In the latter case multipathing global phases (P, PcP, etc.) were efficiently tracked through a heterogeneous earth.

Ray paths associated with later arrivals follow different trajectories to first arrival ray paths. Consequently, they carry different structural information from the regions they probe. It is important to note that later arrivals can be generated by variations in interface geometry in addition to smooth changes in velocity. The structure sampled by a later arriving ray path not only influences the travel time but also manifests in the amplitude and phase shifts across interfaces. All this information can be incorporated into a synthetic seismogram, which here has been done using the Gaussian beam method (e.g. Červený et al., 1982; Červený & Pšenčík, 1984). The ray paths obtained by the Lagrangian wavefront tracker are used to aim narrow fans of rays at the surface near the receiver. These rays are incorporated in the Gaussian beam method to compute synthetic seismograms (see section 4.2) and the impulse response of structure beneath a receiver from teleseismic arrivals (i.e. a receiver function).

The identification of later arrivals in recorded seismic wavetrains is a major obstacle to their use in any application involving real data. Synthetic seismograms, as computed in this work, have the potential to be useful in this regard as they contain more information than just arrival times. For example, they could be used in some kind of waveform inversion approach in receiver function analysis. The potential advantage here is that the synthetic waveform would properly account for lateral variations in structure, unlike conventional receiver function techniques, which assume local one dimensional structures beneath each receiver (e.g. Ammon & Zandt, 1993; Tomlinson et al., 2006).

Presuming that it is possible to extract later arrival travel times from seismograms, they could then be used in travel time tomography to obtain improved results. This is investigated in chapter 5. Synthetic tests clearly show that velocity and interface structure can be recovered more accurately when later arrivals are included compared to when only first arrivals are used. However, if later arrivals are included in seismic tomography, the inverse problem becomes much more non-linear and tends to exhibit multiple local minima. An iterative non-linear procedure may therefore fail to find the global minimum (see figure 5.19). On the other hand, if later arrivals are used, the global minimum of the misfit function is better defined. A two step procedure, in which the solution model is initially obtained using only first arrivals, before adding later arrivals in a second inversion step, may be the best approach. However, in general it is still possible that the solution obtained from first arrivals only may not be sufficiently close to the global minimum to allow the effective inclusion of later arrivals. The solution to the non-linearity of the multi arrival seismic tomography problem therefore might lie in the use of a direct search method (e.g. Gill et al., 1981; Kirkpatrick et al., 1983; Whitley, 1994). One benefit of these non-linear techniques is that there would be no need to calculate ray paths, which would decrease the computation time of the forward problem. Even so, they would only be practical for problems with up to several hundred unknowns; beyond this, direct search methods become computationally unfeasible. Observations of surface wave multipathing are common (e.g. Capon, 1971) and given that the number of unknowns in surface wave tomography can be relatively small (e.g. Fishwick et al., 2005), it therefore might be easier to exploit surface wave data than body wave data.

The focus throughout this work has predominantly been on later arrivals in two dimensions. In chapter 6, a Lagrangian scheme is presented for the computation of multi valued travel times in three dimensions. The wavefront in this case is described by a set of triangles. The surface refinement and simplification algorithms used to model the wavefront were first suggested in the field of computer graphics. The examples demonstrate that the new approach is robust even in complex structures where the wavefront develops secondary and tertiary swallowtails (see figure 6.28). Application to the SEG/EAGE salt dome model (figure 6.28) helps to reinforce its credentials as a viable alternative to previous three dimensional wavefront construction techniques (e.g. Lucio et al., 1996; Vinje et al., 1996a, 1999), which use less sophisticated surface refinement techniques and no surface simplification. Even for a relatively simple velocity structure, the application of surface simplification after every few iterations can lead to a decrease in computation time of 30 % (see section 6.1.3). Further work would be required to extend the three dimensional scheme so that it could be used for the wide variety of problems to which the two dimensional version has been successfully applied in numerical experiments.

The next logical step with the two dimensional wavefront construction technique is to use it in conjunction with real data. The two most promising applications would probably be receiver function analysis and surface wave tomography/ambient noise tomography. In receiver function analysis, the impulse response function may contain later arrivals due to lateral variations in structure. If these later arrivals can be predicted, they could be used as additional constraints. As shown in section 3.2.5, models retrieved for the Australian continent by ambient noise tomography show severe multipathing. This multipathing can be observed in the long term cross correlation of the array data (Saygin, 2007). If this information could be extracted, then it would present an ideal opportunity to evaluate the practical implementation of the new multi arrival tomography technique.

Bibliography

Adalsteinsson, D. & Sethian, J. A., 1995. A fast level set method for propagating interfaces, *J. Comp. Phys.*, **118**(2), 269–277.

Aki, K. & Richards, P. G., 2002. *Quantitative seismology*, University Science Books, 2nd edn.

Alkhalifah, T. & Fomel, S., 2001. Implementing the fast marching eikonal solver: spherical versus Cartesian coordinates, *Geophys. Prosp.*, **49**(2), 165–178.

Aminzadeh, F., Burkhard, N., Long, J., Kunz, T., & Duclos, P., 1996. Three dimensional SEG/EAEG models; an update, *The Leading Edge*, **15**(2), 131–134.

Ammar, K., Wittbold, P., & Carrillo, J., 2006. Scalar conservation laws with general boundary condition and continuous flux function, *J. Diff. Eq.*, **228**(1), 111–139.

Ammon, C., 1991. The isolation of receiver effects from teleseismic P waveforms, *Bull. Seism. Soc. Am.*, **81**(6), 2504–2510.

Ammon, C. & Zandt, G., 1993. Receiver structure beneath the southern Mojave Block, California, *Bulletin of the Seismological Society of America*, **83**(3), 737–755.

Aster, R., Borchers, B., & Thurber, C., 2005. *Parameter estimation and inverse problems*, Academic Press.

Bartels, R. H., Beatty, J. C., & Barsky, B. A., 1987. *An introduction to splines for use in computer graphics and geometric modelling*, Morgan Kaufmann, Los Altos.

Benamou, J. D., 1999. Numerical solution of the high frequency asymptotic expansion for the scalar wave equation, *Comm. Pure Appl. Math.*, **52**(11), 1443–1475.

Benamou, J. D., 2003. An introduction to Eulerian geometrical optics (1992-2002), *J. Sci. Comput.*, **19**(1-3), 63–93.

Berger, M. & Colella, P., 1989. Local adaptive mesh refinement for shock hydrodynamics, *J. Comp. Phys.*, **82**(1), 64–84.

Bevc, D., 1997. Imaging complex structures with semirecursive Kirchhoff migration, *Geophysics*, **62**(2), 577–588.

Bijwaard, H. & Spakman, W., 1999. Fast kinematic ray tracing of first- and later-arriving global seismic phases, *Geophys. J. Int.*, **139**(2), 359–369.

Bohlen, T., 2002. Parallel 3-D viscoelastic finite difference seismic modelling, *Computers and Geosciences*, **28**(8), 887–899.

Bondy, J. & Murty, U., 1976. *Graph theory with applications*, Macmillan London.

Bregman, N., Bailey, R., & Chapman, C., 1989. Crosshole seismic tomography, *Geophysics*, **54**(2), 200–215.

Brenders, A. J. & Pratt, R. G., 2006. Full waveform tomography for lithospheric imaging: results from a blind test in a realistic crustal model, *Geophys. J. Int.*, **168**(1), 133–151.

Bryson, S. & Levy, D., 2003. High-order central WENO schemes for multidimensional Hamilton-Jacobi equations, *SIAM J. Num. Anal.*, **41**(4), 1339–1369.

Bulant, P., 1999. Two-point ray-tracing and controlled initial-value ray-tracing in 3-D heterogeneous block structures, *Jrnl. Seismic Explor.*, **8**(1), 57–75.

Bulant, P., 2002. Sobolev scalar products in the construction of velocity models: Application to model Hess and to SEG/EAGE salt model, *Pure Appl. Geophys.*, **159**(7), 1487–1506.

Burchard, B., Cheng, L., Merriman, B., & Osher, S., 2001. Motion of curves in three spatial dimensions using a level set approach, *J. Comp. Phys.*, **170**(2), 720–741.

Buske, S. & Kästner, U., 2004. Efficient and accurate computation of seismic travel times and amplitudes, *Geophys. Prosp.*, **52**(4), 313–322.

Capon, J., 1971. Comparison of Love- and Rayleigh-wave multipath propagation at LASA, *Bull. Seism. Soc. Am.*, **61**(5), 1327–1344.

Catmull, E. & Clark, J., 1978. Recursively generated B-spline surfaces on arbitrary topological meshes, *Comp. Aided Design*, pp. 350–355.

Červený, V., 1987. Ray tracing algorithms in three-dimensional laterally varying layered structures, in *Seismic tomography: With applications in global seismology and exploration geophysics*, edited by G. Nolet, pp. 99–133, D. Reidel, Dordrecht.

Červený, V., 2001. *Seismic ray theory*, Cambridge University Press, 1st edn.

Červený, V. & Firbas, P., 1984. Numerical modelling and inversion of travel times of seismic body waves in inhomogeneous anisotropic media, *Geophys. J. Royal astr. Soc.*, **76**(1), 41–51.

Červený, V. & Hron, F., 1980. The ray series method and dynamic ray tracing system for 3-D inhomogeneous media with curved interfaces, *Bull. Seism. Soc. Am.*, **70**(1), 47–77.

Červený, V. & Pšenčík, I., 1984. Gaussian beams in elastic 2-D laterally varying layered structures, *Geophys. J. Royal astr. Soc.*, **78**(1), 65–91.

Červený, V., Molotkov, I., & Pšenčík, I., 1977. *Ray method in seismology*, Charles University Press, Praha, 1st edn.

Červený, V., Popov, M., & Pšenčík, I., 1982. Computation of wave fields in inhomogeneous media - Gaussian beam approach, *Geophys. J. Royal astr. Soc.*, **70**(1), 109–128.

Chang, Y. C., Hou, T. Y., Merriman, B., & Osher, S., 1996. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows, *J. Comp. Phys.*, **124**(2), 449–464.

Chapman, C., 1985. Ray theory and its extensions - WKBJ and Maslov seismograms, *J. Geophys.*, **58**(1-3), 27–43.

Chapman, C., 2004. *Fundamentals of seismic wave propagation*, Cambridge University Press, Cambridge.

Chen, P., Jordan, T., & Zhao, L., 2007. Full three-dimensional tomography: a comparison between the scattering-integral and adjoint-wavefield methods, *Geophys. J. Int.*, **170**(1), 175–181.

Chen, P., Zhao, L., & Jordan, T., 2007. Full 3D tomography for crustal structure of the Los Angeles region, *Bull. Seism. Soc. Am.*, **97**(4), 1094–1120.

Chen, Y. G. & Giga, Y. amd Goto, S., 1991. Uniqueness and existence of viscosity solutions of generalized mean curvature flow equations, *J. Diff. Geom.*, **33**(3), 749–786.

Cheng, N. & House, L., 1996. Minimum traveltime calculation in 3-D graph theory, *Geophysics*, **61**(6), 1895.

Chopp, D., 1993. Computing minimal surfaces via level set curvature flow, *J. Comp. Phys.*, **106**(1), 77–91.

Cockburn, B., Qian, J., Reitich, F., & Wang, J., 2005. An accurate spectral/discontinuous finite-element formulation of a phase-space-based level set approach to geometrical optics, *J. Comp. Phys.*, **208**(1), 175–195.

Coman, R. & Gajewski, D., 2005. Traveltime computation by wavefront-orientated ray tracing, *Geophys. Prosp.*, **53**(1), 23–36.

Conder, J. A. & Wiens, D. A., 2006. Seismic structure beneath the Tonga arc and Lau back-arc basin determined from joint Vp, Vp/Vs tomography, *Geochem. Geophys. Geosyst.*, **7**.

Constable, S. C., Parker, R. L., & Constable, C. G., 1987. Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data, *Geophysics*, **52**(3), 289–300.

Cormier, V., 1987. Focusing and defocusing of teleseismic P waves by known three-dimensional structure beneath Pahute Mesa, Nevada test site, *Bull. Seism. Soc. Am.*, **77**(5), 1688–1703.

Courant, R., Friedrichs, K., & Lewy, H., 1928. Über die partiellen Differenzengleichungen der mathematischen Physik, *Math. Ann.*, **100**(1), 32–74.

Dahlen, F., Hung, S., & Nolet, G., 2000. Fréchet kernels for finite-frequency traveltimes - I. theory, *Geophys. J. Int.*, **141**(1), 157–174.

de Kool, M., Rawlinson, N., & Sambridge, M., 2006. A practical grid-based method for tracking multiple refraction and reflection phases in three-dimensional heterogeneous media, *Geophys. J. Int.*, **167**(1), 253–270.

Di, Y., Li, R., Tang, T., & Zhang, P., 2007. Level set calculations for incompressible two-phase flows on a dynamically adaptive grid, *J. Sci. Comput.*, **31**(1-2), 75–98.

Dijkstra, E. W., 1959. A note on two problems in connection with graphs, *Numer. Math.*, **1**(1), 269–271.

Du, Z. J. & Foulger, G. R., 1999. The crustal structure beneath the northwest fjords, Iceland, from receiver functions and surface waves, *Geophys. J. Int.*, **139**(2), 419–432.

Dunn, R. & Forsyth, D., 2003. Imaging the transition between the region of mantle melt generation and the crustal magma chamber beneath the southern east pacific rise with short-period Love waves, *J. Geophys. Res.*, **108**(B7), 2352–2360.

Dyn, N. & Levin, D., 1994. The subdivision experience, *Wavelets, Images and Surface Fitting*, pp. 229–244.

Dyn, N., Levin, D., & Gregory, J. A., 1990a. A Butterfly subdivision scheme for surface interpolation with tension control, *ACM Transaction on Graphics*, **9**(2), 160–169.

Dyn, N., Levin, D., & Micchelli, C. A., 1990b. Using parameters to increase smoothness of curves and surfaces generated by subdivision, *Comp. Aided Geometric Design*, **7**(1-4), 129–140.

Dziewonski, A. M., 1984. Mapping the lower mantle: Determination of lateral heterogeneity in P velocity up to degree and order 6, *J. Geophys. Res.*, **89**(NB7), 5929–5952.

Engquist, B. & Osher, S. J., 1980. Stable and entropy-satisfying approximations for transonic flow calculations, *Math. Comp.*, **34**(149), 45–75.

Engquist, B. & Runborg, O., 2003. Computational high frequency wave propagation, *Acta Numerica*, **12**, 181–266.

Ettrich, N. & Gajewski, D., 1996. Wavefront construction in smooth media for prestack depth migration, *Pure Appl. Geophys.*, **148**(3/4), 481–502.

Evans, L. C. & Spruck, J., 1991. Motion of level set by mean curvature I, *J. Diff. Geom.*, **33**(3), 635–681.

Farra, V. & Madariaga, R., 1987. Seismic waveform modelling in heterogeneous media by ray perturbation theory, *J. Geophys. Res.*, **92**(B3), 2697–2712.

Fatemi, E., Engquist, B., & Osher, S., 1995. Numerical solution of the high frequency asymptotic expansion for the scalar wave equation, *J. Comp. Phys.*, **120**(1), 145–155.

Fishwick, S., Kennett, B. L. N., & Reading, A. M., 2005. Contrasts in lithospheric structure within the Australian craton - insights from surface wave tomography, *Earth Planet. Sci. Lett.*, **231**(3-4), 163–176.

Fomel, S. & Sethian, J. A., 2002. Fast-phase space computation of multiple arrivals, *Proceedings of the National Academy of Sciences of the United States of America*, **99**(11), 7329–7334.

Frolkovič, P. & Mikula, K., 2007. High-resolution flux-based level set method, *J. Sci. Comput.*, **29**(2), 579–597.

Garland, M. & Heckbert, P. S., 1997. Surface simplification using quadric error metrics, in *Proceedings of the 24th annual conference on computer graphics and interactive techniques*, pp. 209–216.

Geoltrain, S. & Brac, J., 1993. Can we image complex structures with first arrival travel time?, *Geophysics*, **58**(4), 564–575.

Gibson Jr, R., Durussel, V., & Lee, K., 2005. Modeling and velocity analysis with a wavefront-construction algorithm for anisotropic media, *Geophysics*, **70**(4), T63.

Gill, P. E., Murray, W., & Wright, M. H., 1981. *Practical optimization*, Academic Press, London.

Godunov, S. K., 1959. A finite difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics, *Math. Sbr.*, **47**(3), 271–290.

Gottlieb, S. & Shu, C., 1998. Total variation diminishing Runge-Kutta schemes, *Math. Comp.*, **67**(221), 73–85.

Graeber, F. M., Houseman, G. A., & Greenhalgh, S. A., 2002. Regional teleseismic tomography of the western Lachlan Orogen and the Newer Volcanic Province, southeast Australia, *Geophys. J. Int.*, **149**(2), 249–266.

Graves, R., 1996. Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences, *Bull. Seism. Soc. Am.*, **86**(4), 1091–1106.

Gray, S. H. & May, W. P., 1994. Kirchhoff migration using eikonal equation travel-times, *Geophysics*, **59**(5), 810–817.

Haines, E., 1994. Point in polygon strategies, in *Graphics Gems IV*, edited by P. S. Heckbert, pp. 24–46, AP Professional (Academic Press).

Hale, A., Bourgouin, L., & Mühlhaus, H., 2007. Using the level set method to model endogenous lava dome growth, *J. Geophys. Res.*, **112**, B03213.

Hammer, J. & Langston, C., 1996. Modeling the effect of San Andreas fault structure on receiver functions using elastic 3D finite difference, *Bull. Seism. Soc. Am.*, **86**(5), 1608–1622.

Hammer, P. T. C., Dorman, L. M., Hildebrand, J. A., & Cornuelle, B. D., 1994. Jasper Seamount structure: Seafloor seismic refraction tomography, *J. Geophys. Res.*, **99**(B4), 6731–6752.

Harten, A., Engquist, B., Osher, S., & Chakravarthy, S. R., 1987. Uniformly high order accurate essentially non-oscillatory schemes, III, *J. Comp. Phys.*, **71**(2), 231–303.

Hill, N., 2001. Prestack Gaussian-beam depth migration, *Geophysics*, **66**(4), 1240.

Hole, J. A. & Zelt, B. C., 1995. 3-D finite-difference reflection travel times, *Geophys. J. Int.*, **121**(2), 427–434.

Iyer, H. & Hirahara, K., 1993. *Seismic Tomography: Theory and Practice*, Chapman & Hall, London.

Jiang, G. S. & Peng, D. P., 2000. Weighted ENO schemes for Hamilton-Jacobi equations, *SIAM J. Sci. Comput.*, **21**(6), 2126–2143.

Jiang, G. S. & Shu, C.-W., 1996. Efficient implementation of weighted ENO schemes, *J. Comp. Phys.*, **126**(1), 202–228.

Jiang, G.-S. & Tadmor, E., 1998. Nonoscillatory central schemes for multidimensional hyperbolic conservation laws, *SIAM J. Sci. Comput.*, **19**(6), 1892–1917.

Jordan, T., 1978. A procedure for estimating lateral variations from low-frequency eigenspectra data, *Geophys. J.*, **52**(3), 441–455.

Julian, B. R. & Gubbins, D., 1977. Three-dimensional seismic ray tracing, *J. Geophys.*, **43**(1), 95–113.

Kelly, K., Ward, R., Treitel, S., & Alford, R., 1976. synthetic seismograms: a finite-difference approach, *Geophysics*, **41**(1), 2–27.

Kendall, J. & Thomson, C., 1989. A comment on the form of the geometrical spreading equations, with some numerical examples of seismic ray tracing in inhomogeneous, anisotropic media, *Geophys. J. Int.*, **99**(2), 401–413.

Kennett, B., 1983. *Seismic wave propagation in stratified media*, Cambridge University Press New York.

Kennett, B. L. N., Sambridge, M., & Williamson, P. R., 1988. Subspace methods for large scale inverse problems involving multiple parameter classes, *Geophys. J. Int.*, **94**(2), 237–247.

Kennett, B. L. N., Engdhal, E. R., & Buland, R., 1995. Constraints on seismic velocities in the Earth from travel times, *Geophys. J. Int.*, **122**(1), 108–124.

Kim, S. & Cook, R., 1999. 3-D traveltime computation using second-order ENO scheme, *Geophysics*, **64**(6), 1867–1876.

Kirkpatrick, S., Gelatt, Jr, C. D., & Vecchi, M. P., 1983. Optimization by simulated annealing, *Science*, **220**(4598), 671–680.

Komatitsch, D. & Tromp, J., 1999. Introduction to the spectral element method for three-dimensional seismic wave propagation, *Geophys. J. Int.*, **139**(3), 806–822.

Korenaga, J., Holbrook, W., Kent, G., Keleman, P., Detrick, R., Larsen, H., Hopper, J., & Dahl-Jensen, T., 2000. Crustal structure of the southeast Greenland margin from joint refraction and reflection seismic tomography, *J. Geophys. Res.*, **105**(B9), 21,591–21,614.

Lambaré, G., Virieux, J., Madariaga, R., & Jin, S., 1992. Iterative asymptotic inversion in the accoustic approximation, *Geophysics*, **57**(9), 1138–1154.

Lambaré, G., Lucio, P. S., & Hanyga, A., 1996. Two-dimensional multivalued traveltime and amplitude maps by uniform sampling of a ray field., *Geophys. J. Int.*, **125**(2), 584–598.

Langston, C. & Hammer, J., 2001. The vertical component P-wave receiver function, *Bull. Seism. Soc. Am.*, **91**(6), 1805–1819.

Lay, T. & Wallace, T. C., 1995. *Modern global seismology*, Academic Press.

Lee, K. & Gibson Jr., R., 2007. An improved mesh generation scheme for the wavefront construction method, *Geophysics*, **72**(1), T1–T8.

Leidenfrost, A., Ettrich, N., Gajewski, D., & Kosloff, D., 1999. Comparison of six different methods for calculating traveltimes, *Geophys. Prosp.*, **47**(3), 269–297.

Leung, S. & Qian, J., 2007. Transmission traveltime tomography based on paraxial Liouville equations and level set formulations, *Inverse Problems*, **23**(1), 799–821.

LeVeque, R. J., 1992. *Numerical methods for conservation laws*, Birkhäuser Verlag, Basel.

Li, X. G. & Ulrych, T. J., 1993. Traveltime computation in discrete heterogeneous layered media, *Jrnl. Seismic Explor.*, **2**(4), 305–318.

Lin, C. & Liu, X., 2007. Convex ENO schemes for Hamilton-Jacobi equations, *J. Sci. Comput.*, **31**(1-2), 195–211.

Liu, H., Osher, S., & Tsai, R., 2006. Multi-valued solution and level set methods in computational high frequency wave propagation, *Commun. Comput. Phys.*, **1**(5), 765–804.

Liu, X.-D., Osher, S., & Chan, T., 1994. Weighted essentially non-oscillatory schemes, *J. Comp. Phys.*, **115**(1), 200–212.

Lorensen, E. W. & Harvey, E. C., 1987. Marching cubes: A high resolution 3D surface reconstruction algorithm, *Comput. Graph.*, **21**(4), 163–169.

Losasso, F., Fedkiw, R., & Osher, S., 2006. Spatially adaptive techniques for level set methods and incompressible flow, *Comput. and Fluids*, **35**(10), 995–1010.

Lucio, P. S., Lambaré, G., & Hanyga, A., 1996. 3D multivalued travel time and amplitude maps, *Pure Appl. Geophys.*, **148**(3-4), 449–479.

Lutter, W. J. & Nowack, R. L., 1990. Inversion for crustal structure using reflections from the PASSCAL Ouachita experiment, *J. Geophys. Res.*, **95**(B4), 4633–4646.

Merriman, B., Bence, J. K., & Osher, S. J., 1994. Motion of multiple junctions - a level set approach, *J. Comp. Phys.*, **112**(2), 334–363.

Metcalf, M., Reid, J., & Cohen, M., 2004. *Fortran 95/2003 Explained*, Oxford University Press.

Morelli, A. & Dziewonski, A., 1987. Topography of the core-mantle boundary and lateral homogeneity of the liquid core, *Nature*, **325**(6106), 678–683.

Moser, T. J., 1991. Shortest path calculation of seismic rays, *Geophysics*, **56**(1), 59–67.

Mulder, W., Osher, S., & Sethian, J. A., 1992. Computing interface motion in compressible gas dynamics, *J. Comp. Phys.*, **100**(2), 209–228.

Nakanishi, I., 1985. Three-dimensional structure beneath the Hokkaido-Tohoku region as derived from a tomographic inversion of P-arrival times, *J. Phys. Earth*, **33**(3), 241–256.

Nakanishi, I. & Yamaguchi, K., 1986. A numerical experiment on nonlinear image reconstruction from first-arrival times for two-dimensional island arc structure, *J. Phys. Earth*, **34**(2), 195–201.

Nolet, G., 1987. *Seismic Tomography: With Applications in Global Seismology and Exploration Geophysics*, Springer.

Nowack, R. & Aki, K., 1984. The two-dimensional Gaussian beam synthetic method: testing and application, *J. Geophys. Res.*, **89**(B9), 7797–7820.

Okabe, A., Boots, B., & Sugihara, K., 1992. *Spatial tessellations: concepts and applications of Voronoi diagrams*, John Wiley & Sons, Inc. New York, NY, USA.

Osher, S. & Fedkiw, R., 2003. *Level set methods and dynamic implicit surfaces*, Springer Verlag, 2nd edn.

Osher, S. & Sethian, J. A., 1988. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *J. Comp. Phys.*, **79**(1), 12–49.

Osher, S., Cheng, L. T., Kang, M., Shim, Y., & Tsai, Y. H., 2002. Geometric optics in a phase-space-based level set and Eulerian framework, *J. Comp. Phys.*, **179**(2), 622–648.

Otten, R. & van Ginneken, L., 1989. *The annealing algorithm*, Kluwer, BV Deventer, The Netherlands, The Netherlands.

Pébay, P. & Baker, T., 2003. Analysis of triangle quality measures, *Mathematics of Computation*, **72**(244), 1817–1839.

Peng, D. P., Merriman, B., Osher, S., Zhao, H. K., & Kang, M. J., 1999. A PDE-based fast local level set method, *J. Comp. Phys.*, **155**(2), 410–438.

Pereyra, V., Lee, W. H. K., & Keller, H. B., 1980. Solving two-point seismic-ray tracing problems in a heterogeneous medium. Part 1. a general adaptive finite difference method, *Bull. Seism. Soc. Am.*, **70**(1), 79–99.

Podvin, P. & Lecomte, I., 1991. Finite difference computation of traveltimes in very contrasted velocity models: a massively parallel approach and its associated tools, *Geophys. J. Int.*, **105**(1), 271–284.

Popov, M., 1982. A new method of computation of wave fields using Gaussian beams, *Wave Motion*, **4**, 85–97.

Popovici, A. M. & Sethian, J. A., 2002. 3-D imaging using higher order fast marching traveltimes, *Geophysics*, **67**(2), 604–609.

Press, H. W., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P., 1992. *Numerical Recepies in Fortran*, Cambridge University Press, 2nd edn.

Qian, J., 2006. Approximations for viscosity solutions of Hamilton-Jacobi equations with locally varying time and space grids., *SIAM J. Num. Anal.*, **43**(6), 2371–2401.

Qian, J. & Leung, S., 2004. A level set based Eulerian method for paraxial multivalued traveltimes, *J. Comp. Phys.*, **197**(2), 711–736.

Qian, J. & Leung, S., 2006. A local level set method for paraxial geometrical optics, *SIAM J. Sci. Comput.*, **28**(2), 206–223.

Qian, J. & Symes, W. W., 2002. An adaptive finite-difference method for traveltimes and amplitudes, *Geophysics*, **67**(1), 167–176.

Qian, J., Cheng, L., & Osher, S., 2003. A level set based eulerian approach for anisotropic wave propagations, *Wave Motion*, **37**(4), 365–379.

Rawlinson, N. & Sambridge, M., 2003. Seismic traveltime tomography of the crust and lithosphere, *Advances in Geophysics*, **46**, 81–198.

Rawlinson, N. & Sambridge, M., 2004a. Wave front evolution in strongly heterogeneous layered media using the fast marching method, *Geophys. J. Int.*, **156**(3), 631–647.

Rawlinson, N. & Sambridge, M., 2004b. Multiple reflection and transmission phases in complex layered media using a multistage fast marching method, *Geophysics*, **69**(5), 1338–1350.

Rawlinson, N., Houseman, G. A., & Collins, C. D. N., 2001. Inversion of seismic refraction and wide-angle reflection traveltimes for 3-D layered crustal structure, *Geophys. J. Int.*, **145**(2), 381–401.

Rawlinson, N., Kennett, B. L. N., & Heintz, M., 2006. Insights into the structure of the upper mantle beneath the murray basin from 3-D teleseismic tomography, *Aust. J. Earth Sci.*, **53**, 595–604.

Rawlinson, N., Reading, A. M., & Kennett, B. L. N., 2006. Lithospheric structure of Tasmania from a novel form of teleseismic tomography, *J. Geophys. Res.*, **111**, B02301.

Ruuth, S. J., 1998. Efficient algorithms for diffusion-generated motion by mean curvature, *J. Comp. Phys.*, **144**(4), 603–625.

Sambridge, M., 1990. Nonlinear arrival time inversion: Constraining velocity anomalies by seeking smooth models in 3-D, *Geophys. J. Int.*, **102**(2), 653–677.

Sambridge, M. & Kennett, B. L. N., 1990. Boundary value ray tracing in a heterogeneous medium: A simple and versatile algorithm, *Geophys. J. Int.*, **101**(1), 157–168.

Sambridge, M., Gudmundsson, O., Kennett, B. L. N., Price, J. C., & Hood, C., 1993. Surface-wave tracking for complex structures, in *Research School of Earth Sciences (ANU) - Annual Report*, pp. 17–18.

Saygin, E., 2007. *Seismic receiver and noise correlation based studies in Australia*, Ph.D. thesis, Research School of Earth Sciences, The Australian National University.

Schmalzl, J. & Loddoch, A., 2003. Using subdivision surfaces and adaptive surface simplification algorithms for modeling chemical heterogeneities in geophysical flows, *Geochem. Geophys. Geosyst.*, **4**(9), 1–17.

Schneider, W. A.and Ranzinger, K. A., Balch, A. H., & Kruse, C., 1992. A dynamic-programming approach to 1st arrival traveltime computation in media with arbitrarily distributed velocities, *Geophysics*, **57**(1), 39–50.

Serna, S. & Qian, J., 2006. Fifth-order weighted power-ENO schemes for hamilton-jacobi equations, *J. Sci. Comput.*, **29**(1), 57–81.

Sethian, J. A., 1987. Numerical methods for propagating fronts, in *Variational methods for free surface interfaces*, edited by P. Concus & R. Finn, Springer Verlag.

Sethian, J. A., 1999. *Level set methods and fast marching methods*, Cambridge University Press, Cambridge, 2nd edn.

Sethian, J. A. & Popovici, A. M., 1999. 3-D traveltime computation using the fast marching method, *Geophysics*, **64**(2), 516–523.

Shaw, P. R. & Orcutt, J. A., 1985. Waveform inversion of seismic refraction data and applications to young Pacific crust, *Geophys. J.*, **82**(3), 375–414.

Sheriff, R. E. & Geldart, L. P., 1995. *Exploration seismology*, Cambridge University Press, 2nd edn.

Shin, C., Ko, S., Kim, W., Min, D., Yang, D., Marfurt, K., Shin, S., Yoon, K., & Yoon, C., 2003. Traveltime calculations from frequency-domain downward-continuation algorithms, *Geophysics*, **68**(4), 1380–1388.

Shu, C. W. & Osher, S., 1988. Efficent implementation of essentially non-oscillatory shock-capturing schemes, *J. Comp. Phys.*, **77**(2), 439–471.

Shu, C. W. & Osher, S., 1989. Efficent implementation of essentially non-oscillatory shock-capturing schemes, II, *J. Comp. Phys.*, **83**(1), 32–78.

Sobel, P. A. & von Seggern, D. H., 1978. Applications of surface-wave ray tracing, *Bull. Seism. Soc. Am.*, **68**(5), 1359–1379.

Sod, G. A., 1985. *Numerical methods in fluid dynamics*, Cambridge University Press.

Stacy, S. & Nowack, R., 2002. Modeling of wide-angle seismic attributes using the Gaussian beam method, *Studia Geophys. Geod.*, **46**(4), 667–690.

Steck, L. K., Thurber, C. H., Fehler, M., Lutter, W. J., Roberts, P. M., Baldridge, W. S., Stafford, D. G., & Sessions, R., 1998. Crust and upper mantle P wave velocity structure beneath Valles caldera, New Mexico: Results from the Jemez teleseismic tomography experiment, *J. Geophys. Res.*, **103**(B10), 24,301–24,320.

Sun, Y., 1992. Computation of 2D multiple arrrival traveltime fields by an interpolative shooting method, *SEG Technical Program Expanded Abstracts*, pp. 1320–1323.

Sussman, S., Smereka, P., & Osher, S. J., 1994. A level set approach for computing solutions to incompressible two-phase flow, *J. Comp. Phys.*, **114**(1), 146–159.

Tape, C., Liu, Q., & Tromp, J., 2007. Finite-frequency tomography using adjoint methods - methodology and examples using membrane surface waves, *Geophys. J. Int.*, **168**(3), 1105–1129.

Thurber, C. H., 1983. Earthquake locations and three-dimensional crustal structure in the Coyote Lake area, central California, *J. Geophys. Res.*, **88**(B10), 8226–8236.

Thurber, C. H. & Ellsworth, W. L., 1980. Rapid solution of ray tracing problems in heterogeneous media, *Bull. Seism. Soc. Am.*, **70**(4), 1137–1148.

Tomlinson, J. P., Denton, P., Maguire, P. K. H., & Booth, D. C., 2006. Analysis of the crustal velocity structure of the British Isles using teleseismic receiver functions, *Geophys. J. Int.*, **167**(1), 223–237.

Toomey, D. R., Solomon, S. C., & Purdy, G. M., 1994. Tomographic imaging of the shallow crustal structure of the East Pacific Rise at 9°30′N, *J. Geophys. Res.*, **99**(B12), 24,135–24,157.

Toomey, D. R., Wilcock, W. S. D., Solomon, S. C., Hammond, W. C., & Orcutt, J. A., 1998. Mantle seismic structure beneath the MELT region of the East Pacific rise from P and S wave tomography, *Science*, **280**(5367), 1224–1227.

Tromp, J., Tape, C., & Liu, Q., 2005. Seismic tomography, adjoint methods, time reversal and banana-doughnut kernels, *Geophys. J. Int.*, **160**(1), 195–216.

van Trier, J. & Symes, W. W., 1991. Upwind finite-difference calculation of traveltimes, *Geophysics*, **56**(6), 812–821.

Velis, D. R. & Ulrych, T. J., 1996. Simulated annealing two-point ray tracing, *Geophys. Res. Lett.*, **23**(2), 201–204.

Velis, D. R. & Ulrych, T. J., 2001. Simulated annealing ray tracing in complex three-dimensional media, *Geophys. J. Int.*, **145**(2), 447–459.

Versteeg, R. J., 1993. Sensitivity of prestack depth migration to the velocity model, *Geophysics*, **56**(6), 873–882.

Vidale, J. E., 1988. Finite-difference calculation of traveltimes, *Bull. Seism. Soc. Am.*, **78**(6), 2062–2076.

Vidale, J. E., 1990. Finite-difference calculation of traveltimes in 3 dimensions, *Geophysics*, **55**(5), 521–526.

Vinje, V., Iversen, E., & Gjøystdal, H., 1993. Travel-time and amplitude estimation using wave-front construction, *Geophysics*, **58**(8), 1157–1166.

Vinje, V., Iversen, E., Åstebøl, K., & Gjøystdal, H., 1996a. Estimation of multivalued arrivals in 3D models using wavefront construction - Part I, *Geophys. Prosp.*, **44**(5), 819–842.

Vinje, V., Iversen, E., Åstebøl, K., & Gjøystdal, H., 1996b. Estimation of multivalued arrivals in 3D models using wavefront construction - Part II tracing and interpolation, *Geophys. Prosp.*, **44**(5), 843–858.

Vinje, V., Åstebøl, K., Iversen, E., & Gjøystdal, H., 1999. 3-D ray modeling by wavefront construction in open models, *Geophysics*, **64**(6), 1912–1919.

Virieux, J. & Farra, F., 1991. Ray tracing in 3-D complex isotropic media: An analysis of the problem, *Geophysics*, **56**(12), 2057–2069.

Wang, Z. & Dahlen, F., 1995. Spherical-spline parametrization of three-dimensional Earth models, *Geophysical Research Letters*, **22**(22), 3099–3102.

Weber, M., 1988. Computation of body-wave seismograms in absorbing 2-D media using the Gaussian beam method: comparison with exact methods, *Geophys. J. Royal astr. Soc.*, **92**(1), 9–24.

White, D. J., 1989. Two-dimensional seismic refraction tomography, *Geophys. J.*, **97**(2), 223–245.

Whitley, D. L., 1994. A genetic algorithm tutorial, *Stats. and Comput.*, **4**(2), 65–85.

Williamson, P. R., 1990. Tomographic inversion in reflection seismology, *Geophys. J. Int.*, **100**(2), 255–274.

Woodhouse, J. H. & Dziewonski, A. M., 1984. Mapping the upper mantle: Three dimensional modeling of Earth structure by inversion of seismic waveforms, *J. Geophys. Res.*, **89**(NB7), 5953–5986.

Xu, S. & Lambaré, G., 2004. Fast migration/inversion with multivalued rayfields: Part 1 - Method, validation test, and application in 2D to Marmousi, *Geophysics*, **69**(5), 1311–1319.

Xu, S., Lambaré, G., & Calandra, H., 2004. Fast migration/inversion with multivalued rayfields: Part 2 - Applications to the 3D SEG/EAGE salt model, *Geophysics*, **69**(5), 1320–1328.

Zelt, C. A. & Ellis, R. M., 1988. Practical and efficent ray tracing in two-dimensional media for rapid traveltime and amplitude forward modelling, *Can. J. Explor. Geophys.*, **24**(1), 16–31.

Zelt, C. A. & Smith, R. B., 1992. Seismic traveltime inversion for 2-D crustal velocity structure, *Geophys. J. Int.*, **108**(1), 16–34.

Zhang, Y. T. & Shu, C.-W., 2003. High-order WENO schemes for Hamilton Jacobi equations on triangular meshes, *SIAM J. Sci. Comput.*, **24**(3), 1005–1030.

Zhao, D., Kanamori, H., Negishi, H., & Wiens, D., 1996a. Tomography of the source area of the 1995 Kobe earthquake: Evidence for fluids at the hypocenter?, *Science*, **274**(5294), 1891–1894.

Zhao, H., Chan, T., Merriman, B., & Osher, S., 1996b. A variational level set approach to multiphase motion, *J. Comp. Phys.*, **127**(1), 179–195.

Zorin, D., Schroeder, P., & Sweldens, W., 1996. Interpolating subdivision for meshes with arbitrary topology, in *Proceedings of the 23rd annual conference on computer graphics and interactive techniques*, pp. 189–192.

Appendix A

Glossary

The aim of this appendix is to provide brief definitions for some of the terminology used in this thesis which may be unfamiliar to readers with a background in seismology. In applied mathematics, these terms are in common usage and have been adopted here both for convenience and to acknowledge the important contributions made by applied mathematicians to this field of research. Note that in order to make the definitions fit into a glossary, they should be viewed as indicative, and not mathematically rigorous.

Bicharacteristic strip

In full phase space or reduced phase space the wavefront is commonly referred to as the bicharacteristic strip. The bicharacteristic strip is defined by the phase space representation of the characteristics of the eikonal equation, which correspond to rays in real space.

Eulerian

In this work, Eulerian refers to the use of a fixed underlying grid of points to describe the motion of an object (i.e. a wavefront). One example of an Eulerian description of the path taken by a wavefront, is its arrival time at certain nodes of the grid (i.e. a travel time field).

Fermat's principle

Fermat's principle states that a ray path between two fixed points P and Q is a path of stationary time. Thus, if $u(\mathbf{x})$ defines slowness and s is path length, the travel time of a ray path is given by

$$t_{PQ} = \int_P^Q u(\mathbf{x})\,\mathrm{d}s = \text{extremum}$$

i.e. t_{PQ} corresponds to either a maximum, minimum or saddle point.

Full phase space

The position $\mathbf{x}$ of a particle in real or normal space and a slowness vector $\mathbf{p} = \nabla T$ describing its motion allows phase space to be defined as the space spanned by $(\mathbf{x}, \mathbf{p})$. For two dimensional real space, $(\mathbf{x}, \mathbf{p})$ spans four dimensional phase space.

Hyperbolic conservation law

A partial differential equation for $u(x, t)$ of the form

$$u_t + [G(u)]_x = 0,$$

is known as a hyperbolic conservation law. A simple example is Burgers' equation given by

$$u_t + uu_x = 0,$$

which describes the motion (i.e. velocity u) of a compressible fluid in one dimension. In this case the solution can develop discontinuities (see section 2.2.3.1), known as rarefactions, where the fluid undergoes sudden expansion, and shocks, where it undergoes sudden compression.

Lagrangian

In this work Lagrangian refers to the use of explicit point (or particle) locations to describe some attribute (e.g. wavefront geometry). In wavefront construction, the point locations are the dependent variables.

Manifold

An n dimensional manifold is an abstract mathematical space, where the neighbourhood of a point can be described using an n dimensional Euclidean space. An example of a two dimensional manifold is the surface of the earth which appears to be flat on a small scale but is spherical on a large scale. A line or a curve is an example of a one dimensional manifold. Manifolds are useful because they allow complicate structures to be expressed using the properties of simpler spaces.

Reduced phase space

From full phase space, one can reduce the number of dimensions needed to describe the motion of a particle. The slowness vector ∇T is replaced by one or more angles describing its orientation with respect to the axes of the coordinate system used for $\mathbf{x}$. For a two dimensional real space, one can then define a three dimensional reduced phase space and a four dimensional phase space (see section 2.3).

Signed distance function

A signed distance function describes the position of an interface implicitly. Its value at a certain node of an underlying grid is given by the distance to the closest point on the interface. The sign is used to determine on which side of the interface the node is located. For a closed interface, the signed distance function is typically defined to be negative for points inside and positive for points outside the interface. Thus, a signed distance function provides an Eulerian description for the position of the interface, which is given by its zero iso-contour line or zero level set (see section 2.2.2).

Wavefront construction

In wavefront construction a wavefront is explicitly represented using a set of points. In two dimensions these points form a curve given by a set of line segments and in three dimensions they form a surface given by a set of patches. Local ray tracing is used to update their position in order to advance the wavefront forward in time. Points are dynamically inserted and removed from the wavefront during the propagation process to ensure that their density remains fixed.

Appendix B

Cubic B-spline approximation

Given values at a set of nodes, cubic B-spline approximation provides a piecewise function that is everywhere continuous in the second derivative, both within an interval and at its boundaries. Cubic B-splines are locally supported and do not necessarily intersect their control nodes. The difference between an approximation and an interpolation is that in the latter case the resulting function passes through the control nodes while in the former case it might not pass through all control nodes. The term cubic B-spline approximation is therefore used in this thesis.

In one dimension, if the control nodes are given as c_i, the spline value $B_i(u)$ at an arbitrary position u between the nodes i and $i+1$ is defined by

$$B_i(u) = \sum_{l=-1}^{2} b_l c_{i+l}, \tag{B.1}$$

where $u(0 \leq u \leq 1)$ is expressed as a fraction of the distance between c_i and c_{i+1}. The weighting factors b_l are given by (Bartels et al., 1987)

$$\begin{aligned} b_{-1} &= \frac{1}{6}(1-u)^3, \\ b_0 &= \frac{1}{6}(4 - 6u^2 + 3u^3), \\ b_{+1} &= \frac{1}{6}(1 + 3 + 3u^2 - 3u^3), \\ b_{+2} &= \frac{1}{6}u^3. \end{aligned} \tag{B.2}$$

In two dimensions, with a field defined on a regular grid as $c_{i,j}$, the cubic

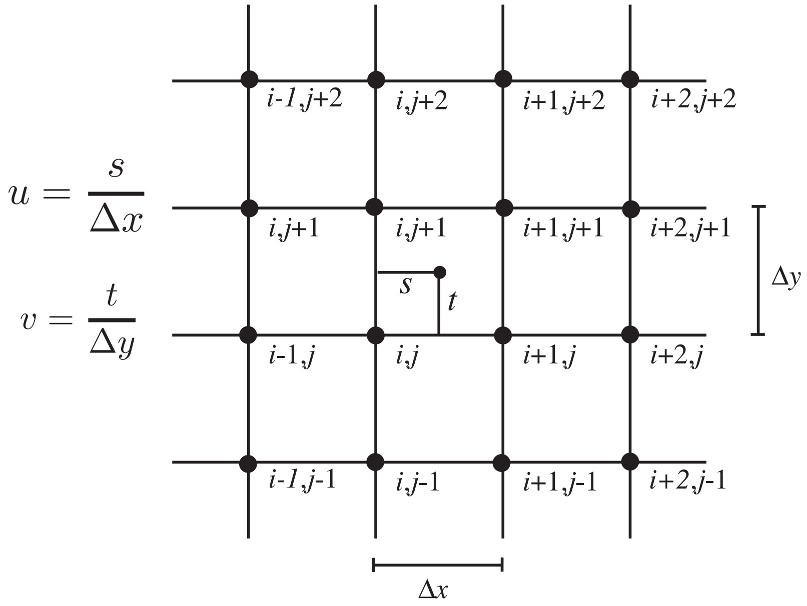


Figure B.1: Cubic B-spline approximation at the arbitrary position (s, t) in the grid cell (i, j). The nodes which are used for the approximation are marked by the black circles.

B-spline value at an arbitrary position (u, v) in the grid cell (i, j) is given by

$$B_{i,j}(u, v) = \sum_{l=-1}^{2} \sum_{m=-1}^{2} b_l b_m c_{i+l,j+m}, \tag{B.3}$$

where u $(0 \leq u \leq 1)$ and v $(0 \leq u \leq 1)$ are expressed as a fraction of the grid spacing (see figure B.1). The weighting factors b_l and b_m are the uniform cubic B-spline functions (B.2).

For the three dimensional case, the value at an arbitrary position (u, v, w) in the grid cell (i, j, k) is given by

$$B_{i,j,k}(u, v, w) = \sum_{l=-1}^{2} \sum_{m=-1}^{2} \sum_{n=-1}^{2} b_l b_m b_n c_{i+l,j+m,k+n}, \tag{B.4}$$

where $u(0 \leq u \leq 1)$, $v(0 \leq v \leq 1)$ and $w(0 \leq w \leq 1)$ are expressed as a fraction of the grid spacing in each of the three orthogonal directions. The weighting factors b_l, b_m and b_n are the uniform cubic B-spline functions (B.2), as before.

By calculating derivatives of the weighting factors with respect to u, v and w, the first and second derivative of the B-spline function can be computed. The first derivatives of the weighting factors are given as

$$\begin{aligned} b'_{-1} &= -\frac{1}{2}(1-u)^2, \\ b'_0 &= \frac{1}{2}(3u^2-4u), \\ b'_{+1} &= \frac{1}{2}(1+2u-3u^2), \\ b'_{+2} &= \frac{1}{2}u^2. \end{aligned} \tag{B.5}$$

The second derivatives of the weighting factors are

$$\begin{aligned} b''_{-1} &= 1-u, \\ b''_0 &= 3u-2, \\ b''_{+1} &= 1-3u, \\ b''_{+2} &= u. \end{aligned} \tag{B.6}$$

In one dimension, if the control nodes are given as c_i, the first derivative of $B_i(u)$ at an arbitrary position u between the nodes i and $i+1$ is given as

$$\frac{\partial}{\partial x}B_i(u) = \frac{1}{\Delta x}\sum_{l=-1}^{2} b'_l c_{i+l}, \tag{B.7}$$

where Δx is the distance between the points i and $i+1$ and the weighting factors b'_l are given by (B.5). The second derivative with respect to x is given by

$$\frac{\partial^2}{\partial x^2}B_i(u) = \frac{1}{\Delta x^2}\sum_{l=-1}^{2} b''_l c_{i+l}, \tag{B.8}$$

where the weighting factors b''_l are given by (B.6).

In figure B.2 a cubic B-spline approximation is applied to an arbitrary set of points in one dimension. As expected, the B-spline function does not pass through the data points, but has a first derivative that is continuous in gradient.

In two dimensions the first derivative of the cubic B-spline function with respect to x is given by

$$\frac{\partial}{\partial x}B_{i,j}(u,v) = \frac{1}{\Delta x}\sum_{l=-1}^{2}\sum_{m=-1}^{2} b'_l b_m c_{i+l,j+m}, \tag{B.9}$$

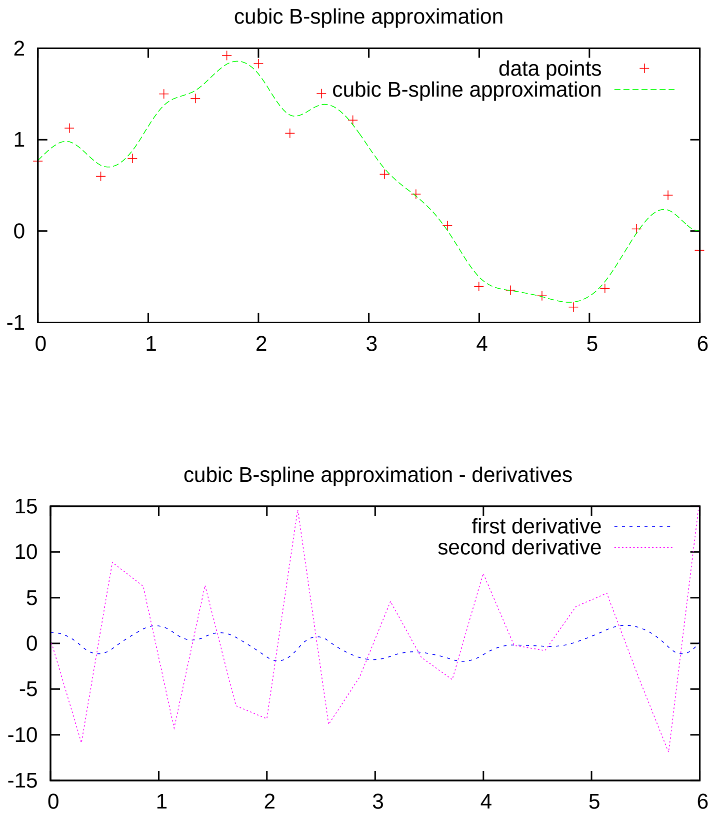


Figure B.2: An example of a cubic B-spline approximation (top) and its first and second derivatives (bottom). The function has been defined by a set of data points. Note how the cubic B-spline curve does not necessarily pass through all the data points, and how the first derivative is smooth and the second derivative is continuous.

and with respect to y by

$$\frac{\partial}{\partial y}B_{i,j}(u,v) = \frac{1}{\Delta y}\sum_{l=-1}^{2}\sum_{m=-1}^{2} b_l b'_m c_{i+l,j+m}. \tag{B.10}$$

The second derivatives in two dimensions are given by

$$\frac{\partial^2}{\partial x^2}B_{i,j}(u,v) = \frac{1}{\Delta x^2}\sum_{l=-1}^{2}\sum_{m=-1}^{2} b''_l b_m c_{i+l,j+m}, \tag{B.11}$$

$$\frac{\partial^2}{\partial y^2}B_{i,j}(u,v) = \frac{1}{\Delta y^2}\sum_{l=-1}^{2}\sum_{m=-1}^{2} b_l b''_m c_{i+l,j+m}, \tag{B.12}$$

$$\frac{\partial^2}{\partial xy}B_{i,j}(u,v) = \frac{1}{\Delta x \Delta y}\sum_{l=-1}^{2}\sum_{m=-1}^{2} b_l b''_m c_{i+l,j+m}. \tag{B.13}$$

Equations equivalent to the above for three or more dimensions can be easily derived. One thing to keep in mind with cubic B-spline approximation is that four nodes in each orthogonal direction are required in order to approximate a value at an arbitrary position in the central cell or segment. This means that a cushion of ghost nodes has to be defined around the edge of the computational domain so that the spline functions completely span the model region.

Appendix C

Runge Kutta scheme

For an ordinary differential equation of the form

$$\frac{\mathrm{d}y}{\mathrm{d}x} = f(x, y), \tag{C.1}$$

a simple formula to advance a discretised solution from x_n to $x_{n+1} = x_n + h$ is given as

$$y_{n+1} = y_n + hf(x_n, y_n). \tag{C.2}$$

This is the so called Euler method, which advances the solution through an interval based on the derivative information at the beginning of the interval. Although simple to implement the Euler method is only accurate to first order. The basic idea behind the Runge Kutta method is to use a step like (C.2) to take a trial step to the midpoint of the interval. The values of x and y at the midpoint are then used to compute the actual step across the whole interval. The equations for a second order Runge Kutta scheme are as follows (e.g. Press et al., 1992):

$$\begin{aligned} k_1 &= hf(x_n, y_n), \\ k_2 &= hf(x + \frac{1}{2}h, y_n + \frac{1}{2}k_1), \\ y_{n+1} &= y_n + k_2 + O(h^3). \end{aligned} \tag{C.3}$$

The fourth order Runge Kutta method is one of the most frequently used variations of this scheme and is given by (e.g. Press et al., 1992):

$$\begin{aligned} k_1 &= hf(x_n, y_n), \\ k_2 &= hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}), \\ k_3 &= hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}), \\ k_4 &= hf(x_n + h, y_n + k_3), \\ y_{n+1} &= y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + O(h^5). \end{aligned} \tag{C.4}$$

Often, Runge Kutta schemes are combined with an adaptive step size control. Those schemes normally use the difference between the solution computed using a higher order scheme and a lower order scheme to determine an estimate of the error. Given a time step Δt, an adaptive scheme will take several smaller time steps in between so that the error of the solution at time $t + \Delta t$ is below a certain threshold. This is especially useful if the time step Δt is large i.e. one is not interested in intermediate results. Adaptive Runge Kutta schemes achieve a predetermined accuracy with minimum computational effort (e.g. Press et al., 1992), and can lead to a significant increase in efficiency.

In this work, wavefronts are isochrons (i.e. contour lines of the travel times), so the time step for each node on the wavefront needs to be the same during the tracking process in order to sample the next wavefront. Therefore it is much more convenient to use a fourth order Runge Kutta scheme with a fixed small time step for the wavefront tracking. However, for solving the dynamic ray tracing equations as part of the Gaussian beam method, an adaptive step size can be used as no intermediate results are needed. Therefore, a Cash Karp Runge Kutta step could be used for solving the dynamic ray tracing equations (Press et al., 1992) as an alternative to a fourth order Runge Kutta scheme.

Appendix D

Fréchet derivatives

A Fréchet derivative in this work is the derivative of the travel time with respect to a model parameter. The model parameter can either be the velocity value of a grid node or the position of an interface control node. Strictly speaking, the term Fréchet derivative should only be used for the derivatives of continuous models; for discrete models the term Gateaux derivative should be used (Shaw & Orcutt, 1985). However it is common practice to refer to **G** as the Fréchet matrix, even if the model is discrete, therefore the expression Fréchet derivative/matrix will be used in the following.

D.1 Fréchet derivative for a velocity node

To first order, the Fréchet derivatives for a continuous velocity field can be written as (e.g. Rawlinson & Sambridge, 2003)

$$\frac{\partial t}{\partial v_n} = -\int_{L(v)} v^{-2} \frac{\partial v}{\partial v_n} \, \mathrm{d}l, \tag{D.1}$$

where v_n is the velocity at a particular node and $\frac{\partial v}{\partial v_n}$ is the change of velocity along the ray with respect to a change in v_n.

For the wavefront tracking, the velocity model is parameterized by a grid of nodes and the wave speed at an arbitrary position is given by cubic B-spline approximation (appendix B). As mentioned previously, the Fréchet matrix contains the derivatives of each ray travel time with respect to each model parameter. Once the ray path has been extracted for an arrival, the Fréchet derivatives for this arrival can be computed by following the ray path.

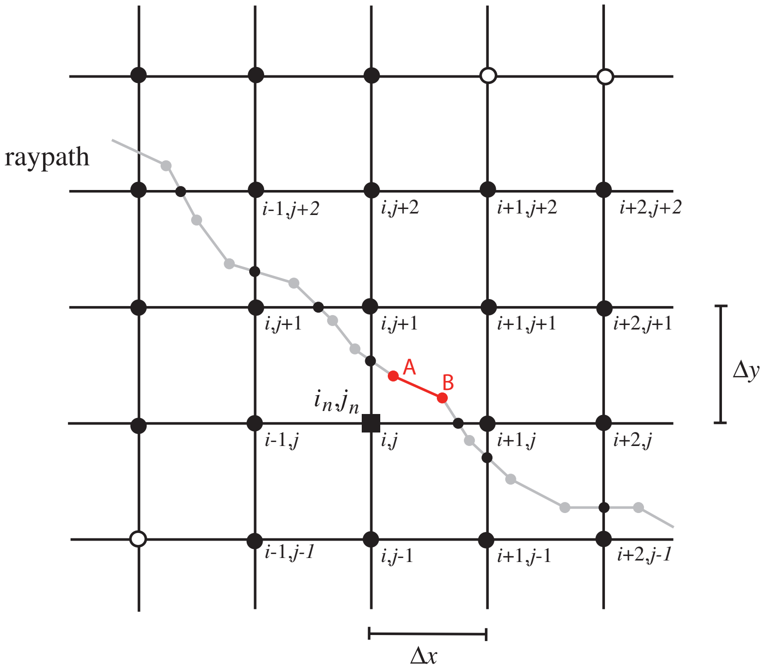


Figure D.1: For a particular ray path the Fréchet derivative for the grid node i_n, j_n (black square) is computed by summing up all the contributions from the different ray path segments in the sixteen surrounding cells in which the velocity is influenced by the value at the grid node i_n, j_n.

A ray path is represented here by several contiguous segments. The length of a segment is given by the separation of two consecutive wavefronts. The length of a ray path segment tends to be smaller than the grid spacing. If a ray path segment spans a cell boundary it is split up into two separate segments (black circles in figure D.1). Figure D.1 shows a ray path for which a contribution to the Fréchet derivative is computed. For a segment of the ray path, like the one marked in red between the point A and B (figure D.1), the derivative of the travel time with respect to a change in the velocity can be formulated. The contribution to the Fréchet derivative at the node i_n, j_n by the the ray path segment between A and B affecting the travel time for the whole ray can be written as

$$\frac{\partial t}{\partial v_{i_n,j_n}} = b_{i_n-i} b_{j_n-j} \frac{\Delta l}{v^{-2}}, \tag{D.2}$$

where Δl is the length of the ray path segment between the endpoints A and B. The weighting functions b_{i_n-i} and b_{j_n-j} are given by (B.2). By following the ray path, the Fréchet derivatives for each ray path segment can be computed. The final Fréchet derivative for a node with respect to the whole ray path is then given

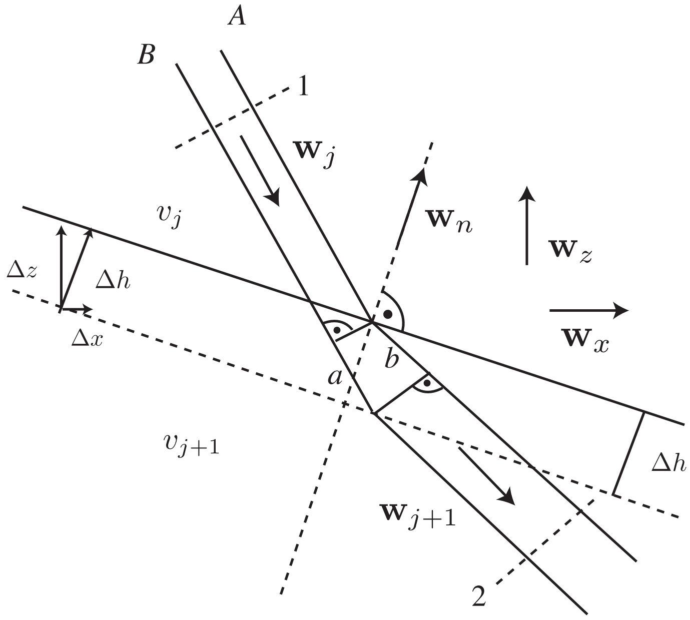


Figure D.2: First order approximation of the Fréchet derivative for a plane wave impinging on a linear interface. $\mathbf{w}_j$ and $\mathbf{w}_{j+1}$ are unit vectors parallel to the rays A and B. $\mathbf{w}_n$ is a unit vector normal to the interface at the intersection point.

by summing up the contribution from the different ray path segments.

D.2 Fréchet derivative for an interface node

When the model parameters describe interface positions, the Fréchet derivatives for a change in the depth coordinate of the model parameter can be written as

$$\frac{\partial t}{\partial z_n} = \frac{\partial t}{\partial h_{int}} \frac{\partial h_{int}}{\partial z_{int}} \frac{\partial z_{int}}{\partial z_n}, \tag{D.3}$$

where z_n is the depth coordinate of the interface node and h_{int} is the displacement in the direction of the interface normal at the intersection point. The first two derivatives on the right hand side of (D.3) can be expressed to first order accuracy using a locally linear interface and wavefront. Figure D.2 shows a plane wave impinging on a planar interface that is perturbed by a distance Δh defined in the direction of the interface normal at the point of the ray intersection. Rays A and B show the path taken by the ray before and after the perturbation of the interface. The difference in travel time between the two rays A and B from position 1 to 2 is

$$\Delta t = \frac{a}{v_j} - \frac{b}{v_{j+1}}. \tag{D.4}$$

The distance a and b in figure D.2 can be expressed using the unit vectors for the ray direction and interface normal as $a = -\mathbf{w}_j \cdot \mathbf{w}_n \Delta h$ and $b = -\mathbf{w}_{j+1} \cdot \mathbf{w}_n \Delta h$. Substituting a and b into (D.4) gives

$$\Delta t = \left[\frac{\mathbf{w}_{j+1} \cdot \mathbf{w}_n}{v_{j+1}} - \frac{\mathbf{w}_j \cdot \mathbf{w}_n}{v_j} \right] \Delta h. \tag{D.5}$$

The approximation to the derivative $\frac{\partial t}{\partial h_{int}}$ can then be written as

$$\frac{\partial t}{\partial h_{int}} \approx \frac{\mathbf{w}_{j+1} \cdot \mathbf{w}_n}{v_{j+1}} - \frac{\mathbf{w}_j \cdot \mathbf{w}_n}{v_j}. \tag{D.6}$$

Using the relationship $\mathbf{w}_n \cdot \mathbf{w}_z = -\frac{\Delta h}{\Delta z}$ the second term in (D.3) can be expressed as

$$\frac{\partial h_{int}}{\partial z_{int}} \approx -\mathbf{w}_n \cdot \mathbf{w}_z. \tag{D.7}$$

Equation (D.3) can then be written as

$$\frac{\partial t}{\partial z_n} \approx \left[\frac{\mathbf{w}_j \cdot \mathbf{w}_n}{v_j} - \frac{\mathbf{w}_{j+1} \cdot \mathbf{w}_n}{v_{j+1}} \right] [\mathbf{w}_n \cdot \mathbf{w}_z] \frac{\partial z_{int}}{\partial z_n}. \tag{D.8}$$

The term $\frac{\partial z_{int}}{\partial z_n}$ depends on the interface depth interpolation or approximation function. If the interface is described using a cubic B-spline approximation the contribution to the Fréchet derivative of the vertical coordinate of the k-th interface node $\mathbf{c}_k$ is expressed as

$$\frac{\partial t}{\partial z_k} \approx \left[\frac{\mathbf{w}_j \cdot \mathbf{w}_n}{v_j} - \frac{\mathbf{w}_{j+1} \cdot \mathbf{w}_n}{v_{j+1}} \right] [\mathbf{w}_n \cdot \mathbf{w}_z] \, b_{k-1}, \tag{D.9}$$

where the weighting function b_{k-i} is given by (B.2). In this case, the intersection point occurs at an arbitrary position between $\mathbf{c}_i$ and $\mathbf{c}_{i+1}$, where $i-1 \leq k \leq i+2$. Similarly one can derive an expression for $\frac{\partial t}{\partial x_n}$. This can be useful if near vertical interfaces are present.

Appendix E

Contents of enclosed CD

Attached to the back of this thesis is a CD, which contains animations for a selection of the wavefront tracking examples shown in chapters 1,2, 3 and 6, as well as a selection of the computer programs developed during the course of this thesis. This appendix briefly describes the contents of the CD. The file `readme.html` located in the root directory of the CD contains similar information.

E.1 Animations

The directory `~/animations` contains nine movies stored as QuickTime files using the H.264 encoder. They can be played using QuickTime (`www.apple.com/quicktime`) or mplayer (`www.mplayer.hu`). The animations refer to examples of propagating wavefronts discussed in different chapters of the thesis using still images.

- **Chapter 1 - Introduction**
  - `~/animations/chapter1/lowan2d.mov`
    Wavefront propagating in the presence of a low velocity anomaly (see figure 1.1). The wavefront has been computed using the Lagrangian approach discussed in chapter 3.
- **Chapter 2 - Eulerian scheme**
  These animations all refer to the example of a plane wave propagating in the presence of a wave guide structure discussed in section 2.5.2.
  - `~/animations/chapter2/epapw51.mov`
    The two signed distance functions are defined on a grid of $51 \times 31 \times 51$

nodes. The grid resolution is not high enough and after a while the wavefront ceases to be represented by a smooth line. Note how the connectivity of the blue surface is no longer preserved as elements of the surface start to lie in the same grid cell.

– `~/animations/chapter2/epapw151.mov`
  The grid consists of 151 × 91 × 151 nodes. The grid resolution has been increased by a factor of three (compared to `epapw51.mov`) and the resulting wavefront is smoother. Ultimately the grid resolution is still not high enough to describe the wavefront adequately during the whole computation, as one can see towards the end of the animation.

– `~/animations/chapter2/plg2000.mov`
  A Lagrangian approach with no refinement or simplification is applied to the same problem. A constant number of 2000 points is used to describe the bicharacteristic strip. The wavefront appears to be tracked much more accurately by the Lagrangian scheme.

- **Chapter 3 - Lagrangian scheme**
  The following animations refer to the two examples presented in sections 3.2.4 and 3.2.5.

  – `~/animations/chapter3/marm.mov`
    Wavefront tracking in the smooth version of the Marmousi model. To obtain the smooth version the original model has been convolved with a spatial Hanning (cos^2) filter of radius 150 m (see figure 3.16).

  – `~/animations/chapter3/sweau_thick.mov`
    Wavefront tracking in a synthetic phase velocity model for fundamental mode Rayleigh waves with a period of 15 s for the east Australian region (see figure 3.19).

- **Chapter 6 - Multi valued travel times in three dimensions**
  The following three animations refer to the low velocity anomaly test case used in section 6.1.3 and the examples presented in sections 6.2.2 and 6.2.3.

  – `~/animations/chapter6/lowan3d.mov`
    Wavefront propagating in the presence of a low velocity anomaly of up to 22 % below the background model. The background velocity distribution is given by 3.0 km/s at the surface and 5.0 km/s at the

bottom and a constant velocity gradient in between (see figure 6.10). The wavefront is coloured according to height.

- `~/animations/chapter6/comp3d.mov`
  A random velocity model with a distinct low velocity region and a source near the surface. The wavefront develops several swallowtails as it propagates. The wavefront is again coloured according to height (see figure 6.24).
- `~/animations/chapter6/salt.mov`
  Wavefront propagating in the smoothed version of the SEG/EAGE salt dome model. Part of the wavefront surface has been removed to facilitate visualisation, and a slice through the velocity model is also plotted (see figure 6.28).

E.2 Programs

The directory `programs` contains a selection of the computer programs developed during the course of this thesis. They are provided as source code together with several examples demonstrating their use. Each program directory contains a `readme` and `makefile`. The `makefile` can be used to build the applications and set up the examples.

- **dnops** - `~/programs/dnops`
  This is an example of a grid based scheme for the computation of first arrival travel times in two dimensions. The program is based on an algorithm described by Kim & Cook (1999). A 5th order WENO scheme is used as the finite difference solver for the eikonal equation. In the first step travel times are computed along the boundaries of an expanding box. In the second step post sweeping is applied to the travel time field. Source grid refinement is also part of the implementation. A single example is provided, where a travel time field is computed for a random velocity model (see figure 1.4).

- **mart2dsmo** - `~/programs/mart2dsmo`
  This is an implementation of the Lagrangian wavefront tracker described in chapter 3, and is the forward solver used for seismic tomography with later arrivals for a smooth velocity model discussed in chapter 5. The program used for solving the inverse step is also included. The directory contains three examples.

  - *Random velocity structure*

    Wavefronts are computed for the random velocity model used in chapter 1 (see figure 1.5).

  - *Marmousi model*

    Wavefront tracking and ray path extraction for the smoothed version of the Marmousi model (section 3.2.4).

  - *Seismic tomography with later arrivals.*

    Compares the results obtained by seismic tomography if (1) only first arrivals and (2) first and later arrivals are used. This numerical test is similar to the one presented in section 5.4.1.

- **mart2dint** - `~/programs/mart2dint`

  This is an implementation of the Lagrangian wavefront tracker described in chapter 4, and is the forward solver used for seismic tomography with later arrivals for a layered velocity model discussed in chapter 5. The directory contains three examples.

  - *Phases for the ak135 global model*

    Wavefront tracking and ray path extraction is performed for P, PcP and $PKiKP$-phases in the ak135 global model (see figures 4.13, 4.15 and 4.17).

  - *Phases in a randomly perturbed ak135 global model*

    Wavefronts and ray paths for P and PcP-phases are computed for the ak135 model with Gaussian distributed random noise added to the velocity field and core mantle boundary topography (see figure 4.19).

  - *Waveform for an incoming teleseismic P-wave*

    A synthetic seismogram is computed using the Gaussian beam method for an incoming teleseismic P-wave and its reverberations due to a structure beneath a receiver (see figures 4.21, 4.22 and 4.24).

- **mart3d** - `~/programs/mart3dsmo`

  Software for the Lagrangian wavefront tracker discussed in chapter 6. The directory contains an example of a wavefront propagating in the presence of a low velocity anomaly. The wavefront develops the three dimensional equivalent of a swallowtail (see figure 6.10).