Journal of Computational Physics **148,** 149–168 (1999)
Article ID jcph.1998.6113, available online at http://www.idealibrary.com on IDEAL®

Notes on Big Ray Tracing

R. H. Rietdijk

Shell International Exploration and Production B.V., Research and Technical Services, Volmerlaan 8, 2288 GD Rijswijk, The Netherlands
E-mail: R.H.Rietdijk@siep.shell.com

Received March 18, 1997; revised May 28, 1998

It is well known that eikonal solvers provide a fast and accurate way to compute traveltimes in complex media. However, they only give solutions corresponding to first arrivals. In (*J. Comput. Phys.* **128**, 463 (1996)), J. D. Benamou presents an algorithm which gives a multivalued traveltime solution on the whole domain considered, by defining subdomains in this domain in such a way that within these subdomains the traveltime is single-valued and can be found by an eikonal solver restricted to the subdomain. The subdomains may overlap, but together they constitute the whole domain. The solution on the whole domain is then given by the combination of the solutions on the subdomains. An approximate realisation of these subdomains, called big rays, is given. In the present paper we describe some problems the algorithm shows and give an explanation of their origin. We conclude that the method cannot be used in the presence of caustics.

1. INTRODUCTION

Numerical methods for modelling wave propagation are an important tool in seismic imaging. One approach is to compute the full wave field by integrating the relevant wave equation using finite element methods, spectral methods, or finite-difference schemes. The result is a grid with, at different times, the wave field given in each point. However, these methods are very slow. Another, much faster, approach is the computation of high-frequency asymptotic solutions. In an acoustic medium of constant density, given a velocity function $c(x)$, the asymptotics of the wavefield is described by the eikonal equation

$$[\nabla\phi(x)]^2 = \frac{1}{c(x)^2}, \tag{1}$$

giving the traveltime function $\phi(x)$, while the amplitude function $A(x)$ is a solution of the

0021-9991/99 $30.00

transport equation

$$\nabla \cdot [A^2(x)\nabla\phi(x)] = 0. \tag{2}$$

These equations can be solved by raytracing or by eikonal solvers. Raytracers shoot rays from a source point in various directions and compute the raypath, together with traveltimes and amplitudes at points along the raypath. One may obtain data between the rays by interpolation.

In order to find traveltimes and amplitudes everywhere in a computational domain, raytracing as such is not satisfactory in cases where there are areas through which very few rays pass, since in these shadow zones interpolation is not reliable. A fast and elegant solution to this problem, the *wavefront construction method*, is described in [2]. The method is based on infill shooting; when the distance or angle between two rays exceeds a certain critical value, an extra ray is started. Its starting position and direction is found by interpolation.

Here we consider another fast method to compute traveltimes in complex velocity models, namely by solving the eikonal equation using finite difference methods. Instead of computing the traveltimes in points along a set of rays, this leads to solutions in each point on a grid in the domain that we consider and, thus, avoids problems in the shadow zones.

Fermat's principle states that a ray always takes a path of extremal traveltime to go from one point to another. Hence, a solution of the eikonal equation at each point in the domain is given by the minimum traveltime from the source to the considered point over all possible paths contained in the domain. This is called the viscosity solution. In the next section we will describe a numerical algorithm proposed in [3] to find the viscosity solution.

The algorithm given there does not always lead to a complete solution. This is caused by the fact that in many velocity models there are points which are connected by more than one ray to the source. Hence, the traveltime function becomes multivalued. The viscosity solution only gives the time of the first arrivals (minimal traveltime) and thus is not complete in many cases. Later arrivals correspond to local traveltime minima, to maxima, or to saddle points. A possible solution to the problem in case of local minima is proposed in [1] and will be explained in Section 3. The idea is that these solutions can in principle be found by defining subdomains in the computational domain, called big rays, in such a way that within these subdomains the traveltime is single-valued. The big rays may overlap, but together they constitute the whole domain. The multivalued solution on the whole domain is then given by the combination of the single-valued solutions on the big rays. In Section 4 we give examples in which multivalued traveltimes are computed using this algorithm. It then becomes clear that the method is not always very accurate. We give an error analysis which indicates under which conditions the method can be used. An explanation of the theoretical background of the problems will be given in Section 5. It is found there that traveltimes computed along a ray which grazes a caustic between the source and the receiver are always (local) maxima in at least one direction. Hence, we conclude that the big ray method cannot be used in the presence of caustics.

2. THE VISCOSITY SOLUTION

In a first-order finite-difference scheme, the derivative of a function in a point only depends on the value of the function in that point and its neighbours on the grid. Therefore the eikonal equation (1) in each point becomes an algebraic equation depending on the

value of ϕ at that point and at its neighbouring points. In [3] a first-order finite-difference scheme is given to find the viscosity solution. We describe this method for two dimensions, but an obvious generalisation to three dimensions exists.

In terms of the slowness $n(x) := 1/c(x)$ the eikonal equation reads

$$\begin{aligned} |\nabla\phi(x)| &= n(x) \quad \text{in } \Omega, \\ \phi(x) &= \phi^{(b)} \quad \text{on } \partial\Omega, \end{aligned} \tag{3}$$

where Ω is an area in the subsurface where the velocity model is given and $\partial\Omega$ is its boundary, with $\phi^{(b)}$ a given boundary condition. In the following we will use the obvious notation

$$A_{ij} := A(x_i, z_j), \quad i = 1, \ldots, M; \ j = 1, \ldots, N, \tag{4}$$

for an arbitrary function $A(x, z)$ on an $M \times N$ grid. Define left and right derivatives in both spatial dimensions as

$$\begin{aligned} D_x^+\phi_{ij} &= \frac{\phi_{i+1,j} - \phi_{ij}}{\Delta x}, \quad D_z^+\phi_{ij} = \frac{\phi_{i,j+1} - \phi_{ij}}{\Delta z}, \\ D_x^-\phi_{ij} &= \frac{\phi_{ij} - \phi_{i-1,j}}{\Delta x}, \quad D_z^-\phi_{ij} = \frac{\phi_{ij} - \phi_{i,j-1}}{\Delta z}. \end{aligned} \tag{5}$$

Furthermore, let the function $g_{ij}(a, b, c, d)$ be defined by

$$g_{ij}(a, b, c, d) = \sqrt{\max[(a^+)^2, (b^-)^2] + \max[(c^+)^2, (d^-)^2]} - n_{ij}, \tag{6}$$

where

$$a^+ := \max(0, a), \quad a^- := \min(0, a). \tag{7}$$

Then a first-order approximation of the eikonal equation is given by

$$\begin{aligned} g_{ij}(D_x^-\phi_{ij}, D_x^+\phi_{ij}, D_z^-\phi_{ij}, D_z^+\phi_{ij}) &= 0 \qquad \text{in } \Omega, \\ \phi_{ij} &= \phi_{ij}^{(b)} \quad \text{on } \partial\Omega. \end{aligned} \tag{8}$$

This can be seen by substituting in (8) the first-order Taylor expansions of the left and right derivatives

$$\begin{aligned} D_x^\pm\phi_{ij} &= \frac{\partial\phi}{\partial x}(x_i, z_j) + \mathcal{O}(\Delta x), \\ D_z^\pm\phi_{ij} &= \frac{\partial\phi}{\partial z}(x_i, z_j) + \mathcal{O}(\Delta z). \end{aligned} \tag{9}$$

When $(\partial\phi/\partial x)(x_i, z_j)$ is positive (resp. negative) the contribution to (8) comes from $D_x^-\phi_{ij}$ (resp $D_x^+\phi_{ij}$). In both cases the contribution equals $(\partial\phi/\partial x)(x_i, z_j) + \mathcal{O}(\Delta x)$. Using similar arguments for $(\partial\phi/\partial z)(x_i, z_j)$ leads to

$$\begin{aligned} &g_{ij}(D_x^-\phi_{ij}, D_x^+\phi_{ij}, D_z^-\phi_{ij}, D_z^+\phi_{ij}) \\ &\quad = \sqrt{\left(\frac{\partial\phi}{\partial x}\right)^2(x_i, z_j) + \left(\frac{\partial\phi}{\partial z}\right)^2(x_i, z_j)} - n_{ij} + \mathcal{O}(\Delta x) = 0, \end{aligned} \tag{10}$$

which is indeed a first-order approximation of the eikonal equation (3). It will become clear

why we choose g_{ij} in this way when we describe some properties of g_{ij} and an algorithm to solve (8) numerically.

1. For fixed values of $\phi_{i+1,j}$, $\phi_{i-1,j}$, $\phi_{i,j+1}$, and $\phi_{i,j-1}$, g_{ij} is a nondecreasing function of ϕ_{ij}. This can be easily understood when we realise that, with increasing ϕ_{ij}, $(D_x^- \phi_{ij})^+$ and $(D_z^- \phi_{ij})^+$ are nonnegative and nondecreasing, while $(D_x^+ \phi_{ij})^-$ and $(D_z^+ \phi_{ij})^-$ are nonpositive and nonincreasing.
2. For fixed values of $\phi_{i+1,j}$, $\phi_{i-1,j}$, $\phi_{i,j+1}$, and $\phi_{i,j-1}$, the values of g in the neighbouring points, $g_{i+1,j}$, $g_{i-1,j}$, $g_{i,j+1}$, and $g_{i,j-1}$ are nonincreasing functions of ϕ_{ij} by a similar argument.
3. $\lim_{\phi_{ij} \to \infty} g_{ij} = \infty$.

Using this, the traveltimes on Ω can be found by the following algorithm.

1. Take $\phi_{ij} = 0$ at the source position and a big value on the boundary of the subsurface area. Big here means at least bigger than the viscosity solution will be. A suitable value can be easily estimated from the maximum value of n over the area. These together form the boundary conditions on $\partial\Omega$ and will be fixed during the process of computation. Furthermore, give ϕ_{ij} a small starting value on Ω, which is the rest of the grid. Here small means at least smaller than the viscosity solution, which in practice leads to the choice $\phi_{ij} = 0$. This implies that everywhere on Ω we have $g_{ij} = -n_{ij}$ to start with.
2. Now update the gridpoints in Ω one by one by computing the value of the traveltime ϕ_{ij} which, together with the traveltime values at neighbouring points, satisfies (8). The fact that only $\max[((D_x^- \phi_{ij})^+)^2, ((D_x^+ \phi_{ij})^-)^2]$ and $\max[((D_z^- \phi_{ij})^+)^2, ((D_z^+ \phi_{ij})^-)^2]$ play a rôle in (8) guarantees that the minimal traveltime (i.e., the traveltime obtained following the fastest route) to each point is computed. This is easily understood when one realises that the contribution from $\max[((D_x^- \phi_{ij})^+)^2, ((D_x^+ \phi_{ij})^-)^2]$ is indeed the contribution from $\min[\phi_{i-1,j}, \phi_{i+1,j}]$ and analogously in the z-direction. As ϕ_{ij} increases, g_{ij} increases from $-n_{ij}$ to zero. However, in neighbouring points g may decrease. Hence, this updating procedure has to be repeated several times in order for the algorithm to converge. It was proven in [3] that it indeed converges to the viscosity solution. The number of steps needed is of the order of the number of gridpoints in one coordinate.

3. BIGRAY TRACING

It was mentioned already in the Introduction that the viscosity solution is not always complete. In many velocity models the traveltime function becomes multivalued because there are points which are connected by more than one ray to the source. All these rays satisfy Fermat's principle of extremal traveltime, although for only one of them this is a global minimum; the other rays correspond to local traveltime minima, to maxima, or to saddle points. The viscosity solution only gives the time of first arrival and, thus, is not complete in many cases. A solution to the problem in case of local minima is described in [1]. It is based on the idea that if there are rays between points which correspond to a local minimum in traveltime when all raypaths in the domain are considered, then it must be possible to find a smaller domain around this path for which the traveltime is an absolute minimum. This small domain around the ray is called the big ray. By covering the whole domain by big rays, computing the viscosity solution in these small domains and combining the results one finds a multivalued solution on the whole domain.

In two dimensions an approximation of the big rays is found by shooting a small number of rays, say 10 or 20, from the source in all directions by using a raytracer; the big rays are then defined as the area between two successive rays [1]. The combination of the solutions in these big rays is an approximation of the multivalued solution of the eikonal on the whole domain. We stress that there is no guarantee that all multivaluedness within a big ray is removed by the above described procedure. There could still be some intersecting rays within the domain and in the point where they intersect the viscosity solution will again only give the first time of arrival. In the next section we will illustrate this with some examples.

4. EXAMPLES

In order to explain the shortcomings of the algorithm we first consider a simple example, found in [1]. It consists of a depth-dependent velocity field, a two-layer model with a smoothed interface (see Fig. 1). We shoot 20 rays from an upper corner of a gridded version of this model (400×200 gridpoints) and define 19 big rays as the gridpoints between two successive rays. We used the traveltimes computed along the rays by the raytracer as boundary conditions to compute the viscosity solution in this big ray (instead of taking a big value at the boundary). The resulting wavefronts are shown in Fig. 4.

Careful inspection shows that the reflected wavefronts are not continuous at the junction between two big rays. This effect is caused by two fundamental shortcomings of the algorithm. To understand this, consider the dark grey bigray in Fig. 2. The two rays which define this big ray intersect and the big ray falls into two separate domains. In the left domain we find part of the incoming wavefront, while in the right domain part of the reflected wavefront is expected to be found. We concentrate on this right part.

The first problem is that at the point S, where the rays intersect there is multivaluedness; both rays arrive at that point at a different time, in general. The algorithm will compute the traveltimes in the right domain taking into account only the lowest of these traveltimes, t_1, as if a point source was switched on at t_1. This leads to traveltimes which are too low, in particular in the upper part of the domain. On the upper limiting ray traveltimes should be computed, starting from the highest traveltime in S, t_2. Hence, the traveltimes computed on this ray are too low by an amount $\Delta t = t_2 - t_1$.

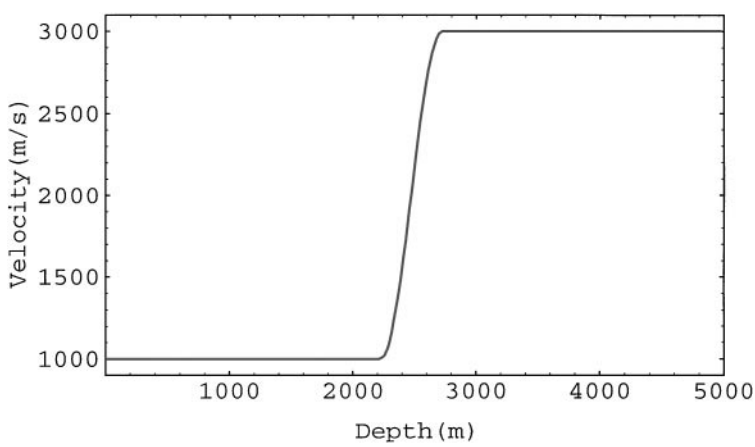

FIG. 1. The velocity as a function of depth for a two-layer model with a smoothed interface.

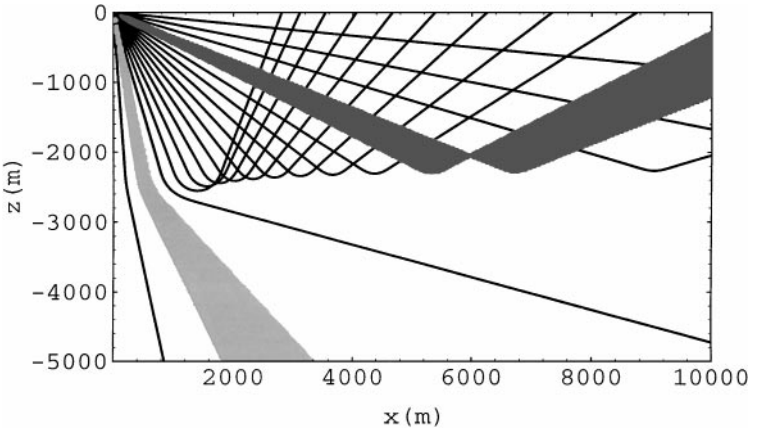

FIG. 2. A raytracer shoots 20 rays. Two big rays are indicated.

On the lower limiting ray, before the point where this ray is reflected, traveltimes are computed which are correct. However, these are not the traveltimes corresponding to the reflected wavefront which should be computed in the right part of the big ray. On this part of the lower limiting ray traveltimes are computed corresponding to the incoming wavefront. After the point where the ray is reflected another effect is visible. In a point R on this part of the ray the algorithm computes the traveltime taking the fastest route from S to R, instead of following the ray. A similar effect plays a role in a point P inside the big ray. In order to compute the correct traveltime a ray should be followed which lies between the two limiting rays and is reflected outside the big ray. The algorithm takes a "short cut"; it arrives in S at time t_1 and takes the fastest route from S to P.

These problems become even more visible when we shoot only 10 rays to start (see Fig. 3), while it seems to almost disappear when more rays are shot (see Fig. 5 for the case of 40 rays). This is to be expected: in the limit where the big rays become smaller $\Delta t \to 0$. Furthermore, the smaller the big rays are the more accurately the algorithm is forced to follow the actual raypath to a point on the reflected wavefront and the smaller the traveltime error is in this point, due to the algorithm taking a faster route. In order to get an idea for how

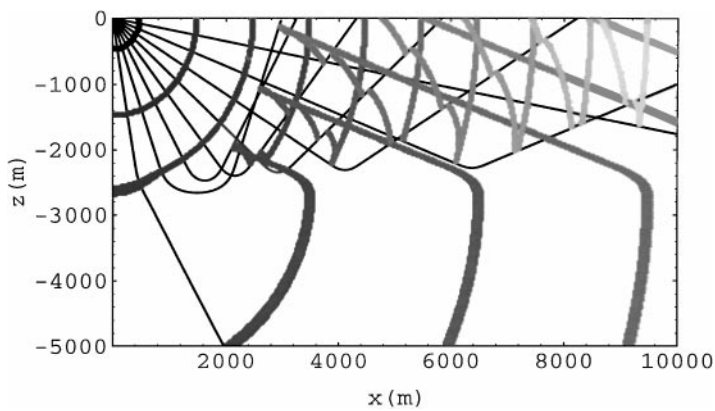

FIG. 3. Wavefronts computed using nine big rays.

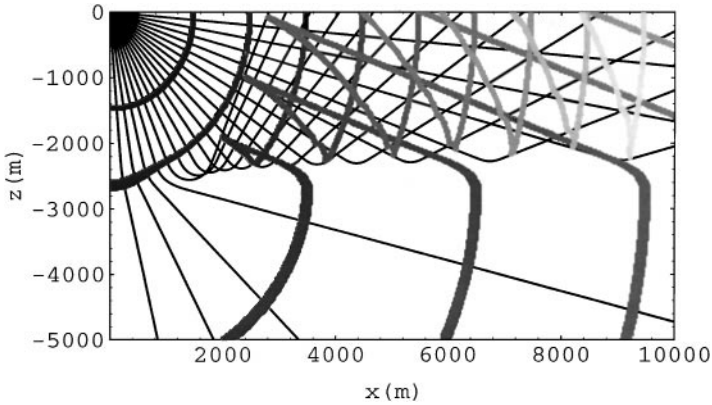

FIG. 4. Wavefronts computed using 19 big rays.

many rays the method becomes reasonably accurate, it seems to be worthwhile to perform an error analysis.

We performed this analysis for a somewhat more general depth-dependent velocity model which we could solve exactly. It consists of two layers with an interface at depth d in which the velocity increases linearly as

$$\begin{aligned} V &= V_1 + k_1 z \qquad \text{for } z < d, \\ V &= V_2 + k_2(z-d) \quad \text{for } z > d, \end{aligned} \tag{11}$$

with

$$V_2 = V_1 + k_1 d, \tag{12}$$

in order for $V(t)$ to be continuous at $z = d$, and with $k_1 < k_2$. Here the positive z-direction points downwards. In the Appendix we analyse this model, compute raypaths and wavefronts, and give exact expressions for the arrival time t_a at the surface as a function of the distance x_a to the source. Rays and wavefronts are computed using the wavefront construction method for $V_1 = 1000$, $d = 2000$, $k_1 = 0.25$, and $k_2 = 1.0$. The result is shown in Fig. 6. In Fig. 7 the traveltime t_a is given as a multivalued function of the offset x_a.

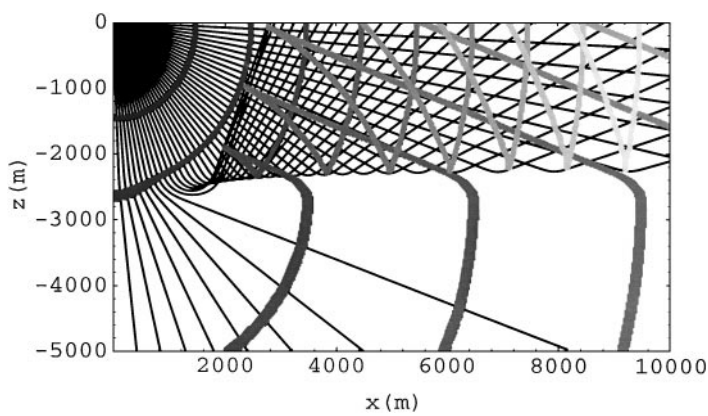

FIG. 5. Wavefronts computed using 39 big rays.

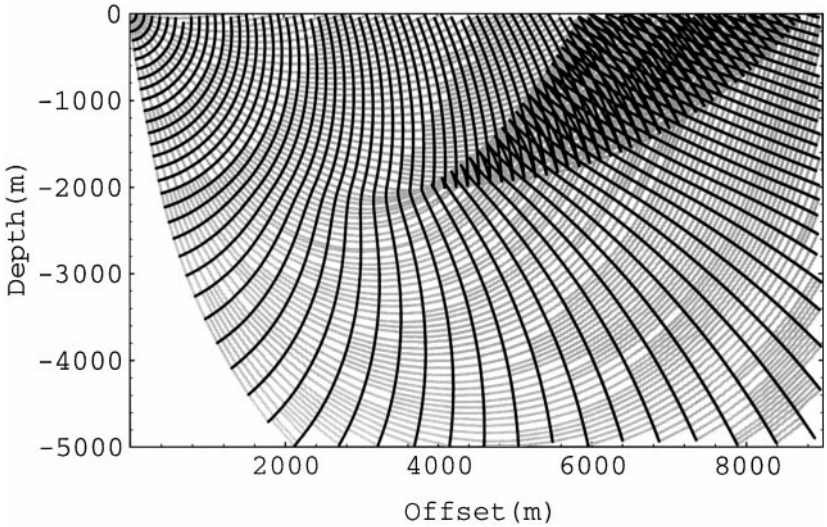

FIG. 6. Rays and wavefronts computed for the velocity model given in Eqs. (11) and (12).

We compared the exact expressions found in the Appendix with numerical results obtained with the big ray method for an increasing number of big rays. The numerical results (grey curve) are plotted, together with the exact results (black curve) in Figs. 8 and 9 for 9 (resp. 39) big rays. There are two types of inaccuracies.

The first one becomes clearly visible when comparing Figs. 8 and 9; when too few big rays are defined, part of the cusp is not found. This is caused by the fact that when the big rays are too big, different raypaths to the same point lie within the same big ray and only the first arrival is found by the algorithm. When the big ray is split up into smaller, partly overlapping bigrays, different raypaths to the same point will be contained in different bigrays and more later arrivals will be found.

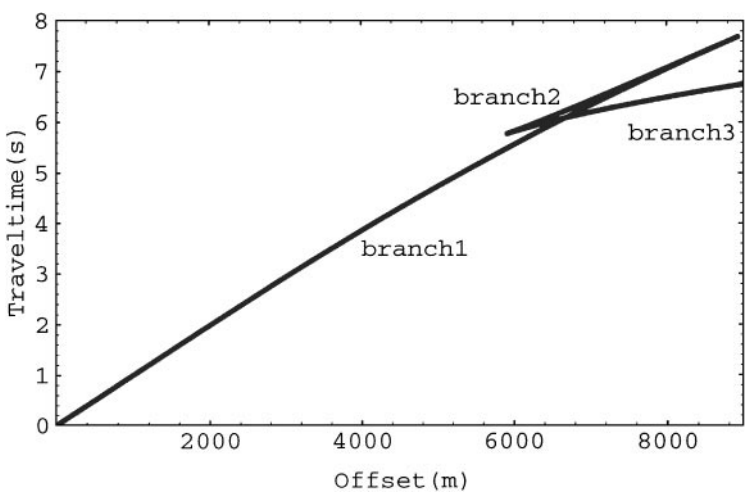

FIG. 7. The arrival time t_a at the surface as a function of the distance x_a to the source for the velocity model given in Eqs. (11) and (12).

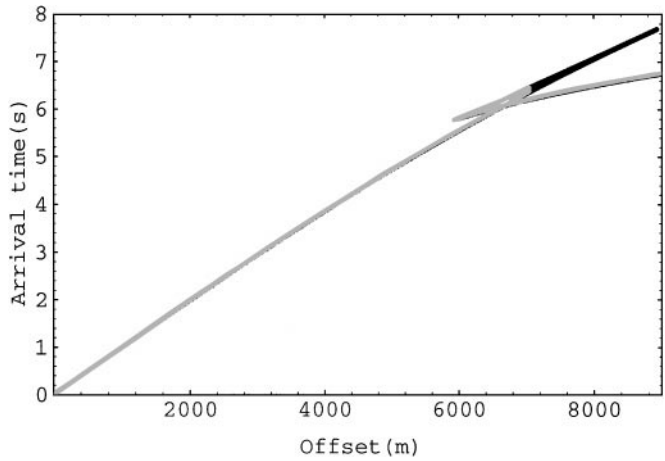

FIG. 8. Black curve: exact solution for $t_a(x_a)$. Grey curve: $t_a(x_a)$ computed using the big ray method with nine big rays, both for the velocity model given in Eqs. (11) and (12).

The other inaccuracy is in the computed traveltimes themselves and is related to the discontinuities in the wavefronts discussed above. We compared the traveltimes computed by the big ray method for different numbers of bigrays, with the exact result. Traveltimes are compared which lie on the same branch of the cusp. As explained above, the big ray method only computes the traveltime corresponding to the first arrival in cases where more than one raypath to a point is contained in the same big ray. In such cases we compared this traveltime with the first arrival computed in the exact model. The result is given in Fig. 10. For more than ± 60 big rays the average relative error

$$\Delta t_{\text{av}} = \left| \frac{t_{\text{comp}} - t_{\text{exact}}}{t_{\text{exact}}} \right|_{\text{av}} \tag{13}$$

converges to a minimal value of 0.25×10^{-3}. For traveltimes up to 8 s and velocities up to

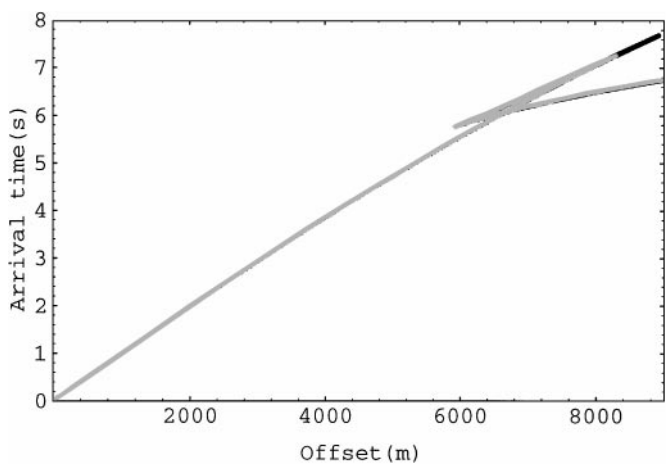

FIG. 9. Black curve: exact solution for $t_a(x_a)$. Grey curve: $t_a(x_a)$ computed using the big ray method with 39 big rays, both for the velocity model given in Eqs. (11) and (12).

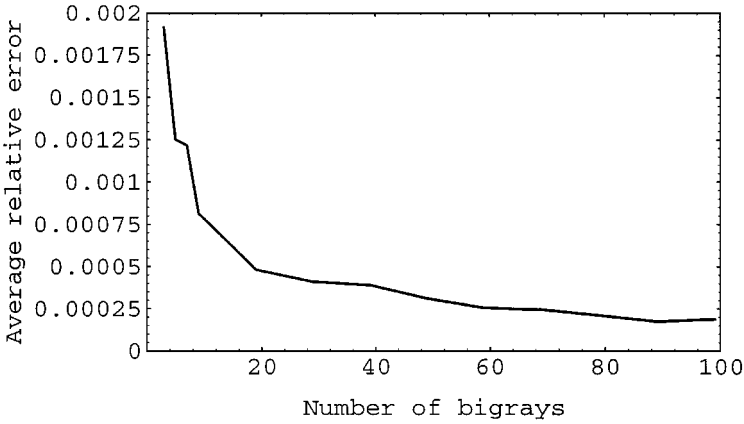

FIG. 10. The average relative error as a function of the number of big rays, for the velocity model given in Eqs. (11) and (12).

4500 m/s this leads to errors up to 9 m, which is usually acceptable for seismic imaging. Notice that the main difference in computational costs of using more big rays is in the initial raytracing. However, this is cheap, compared to finding the viscosity solution within the big ray and, thus, it is advisable to shoot at least 60 rays over 90°.

5. CONJUGATE POINTS AND CAUSTICS

In order to investigate the origin of the problems found in the examples described in the previous section we have a closer look at Fermat's principle. We consider the d-dimensional case in general.

Fermat's principle is a special case of the action principle. The action principle states that a physical path from A to B corresponds to an extremum of the relevant action functional,

$$S[x] = \int_a^b L(x, \dot{x})\, dt, \tag{14}$$

where t is a parametrisation along the path $x_i(t)$ $(i = 1, \ldots, d)$ with $x(a) = A$ and $x(b) = B$. An overdot denotes a derivative with respect to t. In order for the path $x_i(t)$ to correspond to an extremum of $S[x]$, this action should be invariant under all infinitesimal perturbations of the path which vanish at the endpoints. When the position variables $x_i(t)$ are varied in such a way that $\delta x_i(a) = \delta x_i(b) = 0$, the variation of the action reads

$$\begin{aligned} \delta S[\delta x] &= \int_a^b \left\{ L_{x_i} \delta x_i + L_{\dot{x}_i} \delta \dot{x}_i \right\} dt \\ &= \int_a^b \left\{ L_{x_i} - \frac{d}{dt}\left(L_{\dot{x}_i}\right) \right\} \delta x_i \, dt = 0. \end{aligned} \tag{15}$$

Summation over identical indices is implied from 1 to d, unless explicitly indicated otherwise. Equation (15) implies the Euler–Lagrange equations

$$L_{x_i} - \frac{d}{dt}\left(L_{\dot{x}_i}\right) = 0, \tag{16}$$

which describe a physical path.

Fermat's principle states that a ray from a source point A to a receiver in B takes a path of extremal traveltime. The traveltime along a path as a function of the route $x_i(t)$ taken is given by

$$S_0[x] = \int_a^b n(x(t))|\dot{x}|\,dt. \tag{17}$$

It is obvious that the traveltime does not depend on the explicit parametrisation t of the path; $S_0[x]$ is indeed invariant under reparametrisations $t \to \tilde{t}(t)$. We would like to take $S_0[x]$ as the action functional for this problem and derive the ray equations by an action principle.

However, since perturbations of $x_i(t)$ along the path will not give any physical information, we should first remove the reparametrisation invariance in the action. Therefore, we introduce an extra variable $V(t)$ and define a new action functional $S_1[x]$, instead of $S_0[x]$,

$$S_1[x] = \int_a^b \left\{ \frac{1}{2V}|\dot{x}|^2 + \frac{V}{2}n(x)^2 \right\} dt. \tag{18}$$

This action is again reparametrisation invariant when $V(t)$ transforms as a one-form,

$$V(\tilde{t}) = \frac{dt}{d\tilde{t}}V(t). \tag{19}$$

In fact the action $S_1[x]$ represents a whole series of action functionals for different choices of $V(t)$, all describing the same physical problem. The action $S_0[x]$ is obtained from $S_1[x]$ by eliminating $V(t)$, using its Euler–Lagrange equation,

$$V = \frac{|\dot{x}|}{n(x)}. \tag{20}$$

This is allowed because V is not a dynamical field (i.e., its time derivative does not appear in the action). However, by choosing $V(t)$ this way, the reparametrisation invariance is not removed as we want, since the right-hand side of (20) transforms as required by (19). One can remove the reparametrisation invariance in the action by choosing $V(t) = f(t)$ for some $f(t)$ not transforming as a one-form. A reparametrisation $t \to \tilde{t}(t)$ of the action $S[x]$ obtained this way is then equivalent to a different choice for V, $V(\tilde{t}) = (dt/d\tilde{t})\,f(t) =: g(\tilde{t}) \neq f(\tilde{t})$, which leads to a different but gauge-equivalent action. All gauge-equivalent actions of the form (18) for different choices of $V(t)$ can be found from $S[x]$ this way. In these actions the path is labelled by different parameters, but the same physics is described. A convenient gauge condition will turn out to be

$$V = \frac{1}{n(x)^2}. \tag{21}$$

The action then reads

$$S[x] = \int_a^b \left\{ \frac{n^2}{2}|\dot{x}|^2 + \frac{1}{2} \right\} dt =: \int_a^b L(x, \dot{x})\,dt. \tag{22}$$

This is the action we will continue to work with. It is equivalent to $S_1[x]$ and, thus, to $S_0[x]$, if we supplement it explicitly with the Euler–Lagrange equation for V, which does not

follow from it anymore. Equations (20) and (21) together give the constraint

$$|\dot{x}| = \frac{1}{n(x)}. \tag{23}$$

This explains why (21) was a convenient choice for $V(t)$ to fix the reparametrisation invariance; $\dot{x}$ can now be interpreted as the velocity and t as the time-parametrisation along the raypath. Defining the momentum,

$$p_i := L_{\dot{x}_i}, \tag{24}$$

Hamilton's equations are found from the Euler–Lagrange equations,

$$\begin{aligned} \dot{x}_i &= n^{-2} p_i, \\ \dot{p}_i &= n^{-1} \frac{\partial n}{\partial x_i}, \end{aligned} \tag{25}$$

and (23) yields the equation

$$|p| = n(x). \tag{26}$$

A necessary condition for a solution of these equations to correspond to a minimum of the action is that the second variation $\delta^2 S$ is nonnegative for solutions of the Euler–Lagrange equations (16),

$$\begin{aligned} \delta^2 S[\delta x] &= \frac{1}{2} \int_a^b \left\{ L_{x_i x_j} \delta x_i \delta x_j + 2 L_{x_i \dot{x}_j} \delta x_i \delta \dot{x}_j + L_{\dot{x}_i \dot{x}_j} \delta \dot{x}_i \delta \dot{x}_j \right\} dt \\ &= \int_a^b \{ P_{ij} \delta \dot{x}_i \delta \dot{x}_j + Q_{ij} \delta x_i \delta x_j + R_{ij} \delta x_i \delta \dot{x}_j \} \, dt \geq 0. \end{aligned} \tag{27}$$

In the last step we performed a partial integration using $\delta x_i(a) = \delta x_i(b) = 0$ and defined

$$\begin{aligned} P_{ij} &= \frac{1}{2} L_{\dot{x}_i \dot{x}_j}, \\ Q_{ij} &= \frac{1}{2} \left[L_{x_i x_j} - \frac{d}{dt} \left(L_{x_i \dot{x}_j} \right) \right], \\ R_{ij} &= \frac{1}{2} \left[L_{x_i \dot{x}_j} - L_{x_j \dot{x}_i} \right]. \end{aligned} \tag{28}$$

Condition (27) is necessary but not sufficient for $S[x]$ to have a minimum. The action $S[x]$ has a minimum if and only if

$$\delta S[\delta x] = 0 \wedge \delta^2 S[\delta x] = 0 \wedge \cdots \wedge \delta^{2n-1} S[\delta x] = 0 \wedge \delta^{2n} S[\delta x] > 0 \tag{29}$$

for some $n \geq 1$. We consider the simplest case, $n = 1$.

We start by giving the definition of a Jacobi field [5].

DEFINITION 1. Let $x_i(t)$ be an extremal path (i.e., a solution of the Euler–Lagrange equations (16)) from $x_i(a)$ to $x_i(b)$. Define a one-parameter variation,

$$\xi_i(s,t) : (-\epsilon, \epsilon) \times [a,b] \to \mathbf{R}^d \tag{30}$$

of $x_i(t)$, not necessarily keeping the endpoints fixed, with the following properties. The map $\xi_i(s,t)$ is C^∞ and $\xi_i(0,t) = x_i(t)$. Furthermore, for any s_0 fixed, $\xi_i(s_0,t)$ is an extremal path from $\xi_i(s_0,a)$ to $\xi_i(s_0,b)$. A Jacobi field along $x_i(t)$ is then given by the variation vector field,

$$J_i(t) := \frac{\partial \xi_i}{\partial s}(0,t). \tag{31}$$

Consider the extremal $x_i(t)$ and a neighbouring extremal $\xi_i(\delta s, t)$ with infinitesimal δs. Both extremals satisfy the Euler–Lagrange equations (16). Expanding $\xi_i(\delta s, t)$ as a function of δs one finds that the Jacobi field $J_i(t)$ is a solution of the second-order differential equation

$$-\frac{d}{dt}\left[P_{ij}\dot{J}_i + \frac{1}{2}R_{ij}J_i\right] + \frac{1}{2}(Q_{ij} + Q_{ji})\,J_i + \frac{1}{2}R_{ji}\dot{J}_i = 0. \tag{32}$$

This implies that a Jacobi field is completely determined by its initial conditions,

$$J_i(a),\ \frac{dJ_i}{dt}(a). \tag{33}$$

DEFINITION 2. Given an extremal $x_i(t)$ the point $M^* = (a^*, x_i(a^*))$ $(a^* \neq a)$ is said to be conjugate to the point $M = (a, x_i(a))$ if there exists a nonzero Jacobi field $J_i(t)$ along $x_i(t)$ which vanishes for a and a^*.

We can now formulate the following two theorems [6].

THEOREM 1. *The quadratic functional defined in* (27) *is positive* (*negative*) *definite for all variations* δx_i, *satisfying* $\delta x_i(a) = \delta x_i(b) = 0$ *if and only if*:

1. P_{ij} *is a positive* (*negative*) *definite matrix.*
2. $(t, x(t))$ *contains no points conjugate to* $(a, x(a))$ *for* $t \in [a,b]$.

THEOREM 2. *If the quadratic functional defined in* (27) *is nonnegative* (*nonpositive*) *definite for all variations* δx_i *satisfying* $\delta x_i(a) = \delta x_i(b) = 0$ *and if* P_{ij} *is a positive* (*negative*) *definite matrix, then* $(t, x(t))$ *contains no points conjugate to* $(a, x(a))$ *for* $t \in [a,b)$.

We will use these definitions and theorems to check whether $\delta^2 S[\delta x]$ is positive definite. P_{ij} is given by

$$P_{ij} = \frac{n^2}{2}\delta_{ij}. \tag{34}$$

This is indeed a positive definite matrix.

Now we turn to the second condition in Theorem 1; are there points conjugate to a given source point, i.e., is there a one-parameter family of extremals satisfying the conditions given in Definitions 1 and 2? In the following we will construct such a family of extremals.

A raypath $x_i(t)$ is a solution to Hamilton's equations determined by its initial point

$$x_i(a) = A_i \tag{35}$$

and its initial direction

$$\dot{x}_i(a) = \frac{e_i(\boldsymbol{\alpha})}{n(A)}. \tag{36}$$

Here $e(\boldsymbol{\alpha})$ is a unit vector $\in S^{d-1}$ given by the angles $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{d-1})$. Given a source position A_i we parametrise the position space as $x_i(t, \boldsymbol{\alpha})$. A ray is defined as $x_i(t, \boldsymbol{\alpha} = \boldsymbol{\alpha}_0)$, while a wavefront is given by $x_i(t = t_0, \boldsymbol{\alpha})$. This implies

$$\frac{\partial x}{\partial t} \perp \frac{\partial x}{\partial \alpha_j} \quad (j = 1, \ldots, d-1), \tag{37}$$

since rays are orthogonal to wavefronts. The Jacobian of the coordinate transformation is given by

$$\mathcal{J}(t, \boldsymbol{\alpha}) := \begin{vmatrix} \dfrac{\partial x_1}{\partial t} & \dfrac{\partial x_1}{\partial \alpha_1} & \cdots & \dfrac{\partial x_1}{\partial \alpha_{d-1}} \\ \dfrac{\partial x_2}{\partial t} & \dfrac{\partial x_2}{\partial \alpha_1} & \cdots & \dfrac{\partial x_2}{\partial \alpha_{d-1}} \\ \vdots & \vdots & \cdots & \vdots \\ \dfrac{\partial x_d}{\partial t} & \dfrac{\partial x_d}{\partial \alpha_1} & \cdots & \dfrac{\partial x_d}{\partial \alpha_{d-1}} \end{vmatrix}, \tag{38}$$

which is nonzero in general.

DEFINITION 3. A point $x(t_c, \boldsymbol{\alpha}_c)$ is called a caustic point with respect to the source $x(a, \boldsymbol{\alpha}_c)$ if

$$\mathcal{J}(t_c, \boldsymbol{\alpha}_c) = 0. \tag{39}$$

The set of all caustic points is called the caustic set.

For the example described in [1], given in Fig. 5, the caustic set is drawn in Fig. 11.

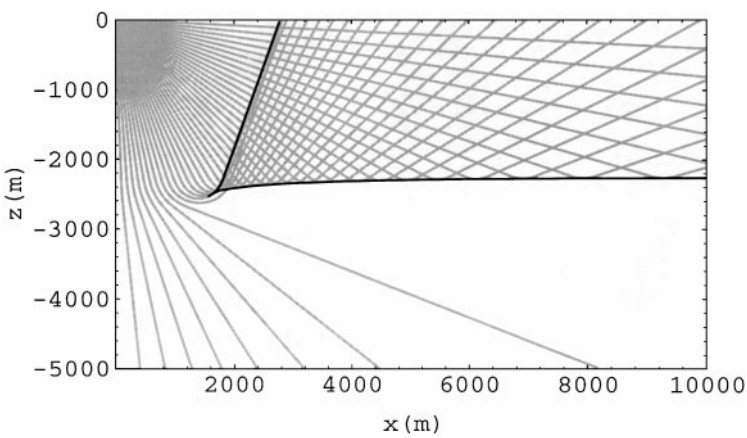

FIG. 11. Rays in grey and caustic line in black for the velocity model given in Fig. 1.

Consider a ray $x_i(t, \boldsymbol{\alpha}_c)$ which grazes the caustic. On the caustic the Jacobian given in (38) vanishes. Hence, in view of (37), there must exist a vector $\delta\boldsymbol{\alpha} \in \mathbf{R}^{d-1}\backslash\{0\}$ for which

$$\sum_{j=1}^{d-1} \frac{\partial x_i}{\partial \alpha_j}(t_c, \boldsymbol{\alpha}_c)\delta\boldsymbol{\alpha}_j = 0. \tag{40}$$

We define a one-parameter family $\xi_i(s, t)$ of neighbouring extremals as a fan of rays shot from the same source $A_i = x_i(a)$ in directions close to $\boldsymbol{\alpha}_c$ in the following way:

$$\xi_i(s, t) = x_i(t, \boldsymbol{\alpha}_c + s\delta\boldsymbol{\alpha}). \tag{41}$$

This defines a Jacobi field

$$J_i(t) = \frac{\partial \xi_i}{\partial s}(0, t) = \sum_{j=1}^{d-1} \frac{\partial x_i}{\partial \alpha_j}(t, \boldsymbol{\alpha}_c)\delta\boldsymbol{\alpha}_j. \tag{42}$$

Close to the source in A, Hamilton's equations can be solved using the initial conditions (35) and (36), giving

$$x_i(t, \boldsymbol{\alpha}) = A_i + (t - a)\frac{e_i(\boldsymbol{\alpha})}{n(A)}. \tag{43}$$

Hence,

$$\left.\frac{\partial x_i}{\partial \alpha_j}\right|_{t=a} = (t - a)\frac{1}{n(A)}\left.\frac{\partial e_i}{\partial \alpha_j}\right|_{t=a} = 0 \tag{44}$$

for $i = 1, \ldots, d$ and $j = 1, \ldots, d-1$. Therefore the Jacobi field $J_i(t)$ vanishes at $t = a$. It also vanishes on the caustic because of (40), but it is nonzero in general since $\mathcal{J}(t, \boldsymbol{\alpha})$ is nonzero in general. The fact that there exists a nonzero Jacobi field which vanishes for $t = a$ and for $t = t_c$ implies that all causic points $x(t_c, \boldsymbol{\alpha}_c)$ are conjugate to the source point $x(a, \boldsymbol{\alpha}_c)$.

The reverse is also true: all points $x(a^*, \boldsymbol{\alpha}_0)$ on a ray $x(t, \boldsymbol{\alpha}_0)$, which are conjugate to $x(a, \boldsymbol{\alpha}_0)$ ($a^* \neq a$) are caustic with respect to $x(a, \boldsymbol{\alpha}_0)$. This is proven as follows. Given that on a ray $x(t, \boldsymbol{\alpha}_0)$ the point $x(a^*, \boldsymbol{\alpha}_0)$ is conjugate to $x(a, \boldsymbol{\alpha}_0)$, Definition 2 says that there is a Jacobi field $J_i^{(0)}(t)$ along $x(t, \boldsymbol{\alpha}_0)$ such that

$$J_i^{(0)}(a) = J_i^{(0)}(a^*) = 0, \tag{45}$$

but

$$\exists t_0 \in (a, a^*) : J_i^{(0)}(t_0) \neq 0. \tag{46}$$

Given that a Jacobi field $J_i(t)$ is completely determined by its initial conditions $J_i(a)$ and $dJ_i/dt(a)$, a basis of Jacobi fields for which $J_i(a)$ vanishes consists of d independent fields $J_i^{(1)}(t), \ldots, J_i^{(d)}(t)$. We construct such a basis explicitly. In a d-dimensional space there are $d-1$ independent directions $\delta\boldsymbol{\alpha}^{(k)}$ ($k = 1, \ldots, d-1$) in which the initial direction of a ray can be varied. Therefore, we can define $d-1$ one-parameter families of rays as

$$\xi_i^{(k)}(s, t) = x_i\left(t, \boldsymbol{\alpha}_0 + s\delta\boldsymbol{\alpha}^{(k)}\right) \quad (k = 1, \ldots, d-1). \tag{47}$$

A dth ray family is defined as

$$\xi_i^{(d)}(s,t) = x_i(t + s(t-a), \boldsymbol{\alpha}_0). \tag{48}$$

From these ray families d Jacobi fields are found

$$\begin{aligned} J_i^{(k)}(t) &= \frac{\partial \xi_i^{(k)}}{\partial s}(0,t) = \sum_{j=1}^{d-1} \frac{\partial x_i}{\partial \alpha_j}(t, \boldsymbol{\alpha}_0)\delta\alpha_j^{(k)} \quad (k = 1, \ldots, d-1), \\ J_i^{(d)}(t) &= \frac{\partial \xi_i^{(d)}}{\partial s}(0,t) = (t-a)\frac{\partial x_i}{\partial t}(t, \boldsymbol{\alpha}_0). \end{aligned} \tag{49}$$

It is straightforward to check that these fields vanish indeed at the source point $x_i(a, \alpha_0)$. Furthermore, they are nonzero and linearly independent in general, since the Jacobian $\mathcal{J}(t, \boldsymbol{\alpha})$ does not vanish except for caustic points.

The Jacobi field $J_i^{(0)}(t)$ can be expressed in terms of this basis

$$J_i^{(0)}(t) = P_{(d)}(t-a)\frac{\partial x_i}{\partial t}(t, \boldsymbol{\alpha}_0) + \sum_{j,k=1}^{d-1} P_{(k)} \frac{\partial x_i}{\partial \alpha_j}(t, \boldsymbol{\alpha}_0)\delta\alpha_j^{(k)} \tag{50}$$

for some set of parameters $P_{(i)}$ $(i = 1, \ldots, d)$. We know from (45) that $J_i^{(0)}(a^*)$ vanishes. This implies, together with (37), that

$$P_{(d)} = 0 \tag{51}$$

and

$$\sum_{j=1}^{d-1} \frac{\partial x_i}{\partial \alpha_j}(a^*, \boldsymbol{\alpha}_0)\delta\tilde{\alpha}_j := \sum_{j,k=1}^{d-1} P_{(k)} \frac{\partial x_i}{\partial \alpha_j}(a^*, \boldsymbol{\alpha}_0)\delta\alpha_j^{(k)} = 0. \tag{52}$$

From this last equation we find

$$\mathcal{J}(a^*, \boldsymbol{\alpha}_0) = 0. \tag{53}$$

Hence $x(a^*, \boldsymbol{\alpha}_0)$ is caustic with respect to the source in $x(a, \boldsymbol{\alpha}_0)$. This leads to the following theorem.

THEOREM 3. *Given a ray $x(t, \boldsymbol{\alpha})$, a point $x(a^*, \boldsymbol{\alpha})$ is conjugate to $x(a, \boldsymbol{\alpha})$ $(a^* \neq a)$ if and only if $x(a^*, \boldsymbol{\alpha})$ is caustic with respect to $x(a, \boldsymbol{\alpha})$.*

This theorem leads to the following conclusions:

1. For a raypath ending in a caustic point $(b = a^*)$ there are variations for which $\delta^2 S[\delta x] \leq 0$, according to Theorem 1. There is at least one direction of variation for which $\delta^2 S[\delta x] = 0$ in that case, namely the Jacobi field $J_i^{(0)}(t)$ given in (50),

$$J_i^{(0)}(t) = \sum_{j=1}^{d-1} \frac{\partial x_i}{\partial \alpha_j}(t, \boldsymbol{\alpha}_0)\delta\tilde{\alpha}_j. \tag{54}$$

$J_i^{(0)}(t)$ vanishes for $t = a$ and $t = a^*$, as is given in Eq. (45). Hence, the boundary conditions for the variation $\delta x = J^{(0)}$ are satisfied. Furthermore, $J_i^{(0)}(t)$ is a solution of (32). It then follows that

$$0 = \int_a^{a^*} \left\{ -\frac{d}{dt}\left[P_{ij}\dot{J}_i^{(0)} + \frac{1}{2}R_{ij}J_i^{(0)} \right] + \frac{1}{2}(Q_{ij} + Q_{ji})J_i^{(0)} + \frac{1}{2}R_{ji}\dot{J}_i^{(0)} \right\} J_j^{(0)}\, dt$$

$$= \int_a^{a^*} \left\{ P_{ij}\dot{J}_i^{(0)}\dot{J}_j^{(0)} + R_{ij}J_i^{(0)}\dot{J}_j^{(0)} + Q_{ij}J_i^{(0)}J_j^{(0)} \right\} dt = \delta^2 S\left[J^{(0)}\right]. \tag{55}$$

2. The quadratic functional given in (27) is not positive definite in case of a raypath which contains a caustic point; it follows from Theorem 2 that when there are conjugate points on the raypath from a to b ($b > a^*$) then there are variations for which $\delta^2 S[\delta x] < 0$. In both this case and in the case when $b = a^*$ one can derive that for variations along the path, i.e., $\delta x = f(t)\dot{x}$ with $f(a) = f(b) = 0$ and $f(t) \not\equiv 0$,

$$\delta^2 S[f\dot{x}] = \frac{1}{2}\int_a^b \left(\frac{df}{dt}\right)^2 dt > 0. \tag{56}$$

For the present case ($b > a^*$) this implies that the variation for which $\delta^2 S[\delta x] < 0$ must deform the path itself. *The raypath corresponds to a maximum of the action in the direction of that variation.* Using the big ray method in cases when this happens will not lead to the right result, since this method only deals with (local) minima. Choosing a big ray around a ray corresponding to a traveltime maximum or saddle point and computing the viscosity solution in this big ray, a solution of minimal traveltime is found, which in most cases corresponds to a ray (partly) following the big ray boundary; the ray takes a "short cut" as much as possible. This is precisely what happens with the reflected rays in the example given in [1]. Only by choosing the big rays small in such cases can one make the effect of this error small enough. The incoming and the refracted wavefront are found correctly. The traveltimes on these wavefronts are minima, one of which is local.

Notice that, although the refracted rays cross the caustic C, there are no conjugate points on these rays. Confusion is caused by the fact that a position on C (and positions in the whole area where multivalued traveltimes occur) can be labelled by more then one set of coordinates $(t, \boldsymbol{\alpha})$. On a reflected ray $x(t, \boldsymbol{\alpha}_c)$ lays a caustic point, labelled as $x(t_c, \boldsymbol{\alpha}_c)$, while on a refracted ray $x(t, \boldsymbol{\alpha}_r)$ this same point in space is labelled as $x(t_r, \boldsymbol{\alpha}_r)$, with $(t_r, \boldsymbol{\alpha}_r) \neq (t_c, \boldsymbol{\alpha}_c)$. The point $x(t_c, \boldsymbol{\alpha}_c)$ is conjugate to the source in $x(a, \boldsymbol{\alpha}_c)$, while $x(t_r, \boldsymbol{\alpha}_r)$ is not.

6. CONCLUSIONS

In this paper we studied the big ray tracing algorithm presented in [1] and designed to compute multivalued traveltimes in complex media. Studying some relevant examples, inaccuracies of the algorithm were found and an error analysis was performed. The preliminary conclusion was that the accuracy of the program is acceptable when one uses at least 60 big rays in a 90° shooting angle. The conclusion of our theoretical analysis is more dramatic. The big ray method is based on the idea that all raypaths correspond to (local) minima, which is not always the case, in particular in the presence of caustics. Therefore, although the error becomes acceptable when the big rays are chosen small enough, it does

not seem to be reasonable to present the method as an algorithm to compute multivalued traveltimes.

APPENDIX

In this appendix we analyse a depth-dependent velocity model which can be solved exactly. It consists of two layers with an interface at depth d in which the velocity increases linearly as

$$\begin{aligned} V &= V_1 + k_1 z \qquad \text{for } z < d, \\ V &= V_2 + k_2(z-d) \quad \text{for } z > d, \end{aligned} \tag{57}$$

with

$$V_2 = V_1 + k_1 d, \tag{58}$$

in order for $V(t)$ to be continuous at $z = d$, and with $k_1 < k_2$. Here the positive z-direction points downwards. It is well known that raypaths and wavefronts in linear velocity models are circle arcs, with centre and radius depending on the velocity function and the initial angle (see, e.g. [4, pp. 272–276]).

Let us first consider only the top layer. For a ray shot from surface at (0, 0), with initial angle θ_0 with the positive z-direction, the path and traveltime are given by

$$\begin{aligned} x &= -\frac{V_1}{k_1 \sin\theta_0}(\cos\theta - \cos\theta_0), \\ z &= \frac{V_1}{k_1 \sin\theta_0}(\sin\theta - \sin\theta_0), \\ t &= \frac{1}{k_1} \ln\left(\frac{\tan\theta/2}{\tan\theta_0/2}\right). \end{aligned} \tag{59}$$

All these quantities are parametrised by θ, the angle of the ray with the positive z-direction. The first two expressions, indeed, describe a circle with radius $V_1/(k_1 \sin\theta_0)$ and centre $V_1(\cos\theta_0, -\sin\theta_0)/(k_1 \sin\theta_0)$. Rays and wavefronts are computed using the wavefront construction method for $V_1 = 1000$, $d = 2000$, $k_1 = 0.25$, and $k_2 = 1.0$. The result is shown in Fig. 6.

Using the expressions given above and analogous ones for rays passing through the second layer, one can find expressions for the arrival position and time at the surface. We distinguish between rays which pass only through the first layer and rays which cross the interface. The critical angle for which the ray grazes the second layer is given by

$$\sin\theta_{\mathrm{cr},1} = \frac{V_1}{V_2}, \tag{60}$$

for which

$$x_a = \frac{2}{k_1}\sqrt{V_2^2 - V_1^2}. \tag{61}$$

We find for $\theta_0 > \theta_{\mathrm{cr},1}$

$$\begin{aligned} x_a &= \frac{2V_1}{k_1 \tan\theta_0}, \\ t_a &= \frac{1}{k_1} \ln\left(\frac{1+\cos\theta_0}{1-\cos\theta_0}\right), \end{aligned} \tag{62}$$

while for $\theta_0 < \theta_{\mathrm{cr},1}$

$$\begin{aligned} x_a &= \frac{2V_1}{k_1 \tan\theta_0} - \frac{2V_1(1-(k_1/k_2))}{k_1 \sin\theta_0} \sqrt{1-\left(V_2^2/V_1^2\right)\sin^2\theta_0}, \\ t_a &= \frac{1}{k_1} \ln\left(\frac{1+\cos\theta_0}{1-\cos\theta_0}\right) - \frac{1}{k_1}(1-(k_1/k_2)) \ln\left(\frac{1+\sqrt{1-\left(V_2^2/V_1^2\right)\sin^2\theta_0}}{1-\sqrt{1-\left(V_2^2/V_1^2\right)\sin^2\theta_0}}\right). \end{aligned} \tag{63}$$

The function $x_a(\theta_0)$ given in (63) has a minimum at

$$\sin\theta_{\mathrm{cr},2} = \sqrt{\frac{1-(1-(k_1/k_2))^2}{\left(V_2^2/V_1^2\right)-(1-(k_1/k_2))^2}} \tag{64}$$

for which

$$x_a = \frac{2}{k_1}\sqrt{\left(V_2^2-V_1^2\right)\left(1-(1-(k_1/k_2))^2\right)}. \tag{65}$$

In Fig. 7 the traveltime t_a is given as a multivalued function of the offset x_a. We distinguish three branches in this function: branch 1 corresponds to $\theta_0 > \theta_{\mathrm{cr},1}$; branch 2 corresponds to $\theta_{\mathrm{cr},2} < \theta_0 < \theta_{\mathrm{cr},1}$; branch 3 to $\theta_0 < \theta_{\mathrm{cr},2}$.

In order to compare our numerical scheme to the exact results obtained for this model, we want to compute traveltimes on a grid. Therefore, we eliminated the angle θ_0 from the expressions for arrival position and time and found the following expressions for $t_a(x_a)$:

for $x_a \le (2/k_1)\sqrt{V_2^2-V_1^2}$,

$$t_a = \frac{2}{k_1} \ln\left(\frac{k_1 x_a}{2V_1} + \sqrt{\left(\frac{k_1 x_a}{2V_1}\right)^2 + 1}\right);$$

for $(2/k_1)\sqrt{\left(V_2^2-V_1^2\right)\left(1-(1-(k_1/k_2))^2\right)} \le x_a \le (2/k_1)\sqrt{V_2^2-V_1^2}$,

$$\begin{aligned} t_a = {} & \frac{1}{k_1} \ln\left(\frac{Q_- + k_1^2 x_a^2 + 4(1-(k_1/k_2))^2\left(V_2^2-V_1^2\right)}{Q_- - k_1^2 x_a^2 - 4(1-(k_1/k_2))^2\left(V_2^2-V_1^2\right)}\right) \\ & - \frac{1}{k_1}\left(1-\frac{k_1}{k_2}\right) \ln\left(\frac{Q_- - k_1 x_a R + 4(1-(k_1/k_2))\left(V_2^2-V_1^2\right)}{Q_- + k_1 x_a R - 4(1-(k_1/k_2))\left(V_2^2-V_1^2\right)}\right); \end{aligned} \tag{66}$$

for $x_a \geq (2/k_1)\sqrt{\left(V_2^2 - V_1^2\right)\left(1-(1-(k_1/k_2))^2\right)}$,

$$t_a = \frac{1}{k_1}\ln\left(\frac{Q_+ + k_1^2x_a^2 + 4(1-(k_1/k_2))^2\left(V_2^2-V_1^2\right)}{Q_+ - k_1^2x_a^2 - 4(1-(k_1/k_2))^2\left(V_2^2-V_1^2\right)}\right)$$
$$-\frac{1}{k_1}\left(1-\frac{k_1}{k_2}\right)\ln\left(\frac{Q_+ + k_1x_aR + 4(1-(k_1/k_2))\left(V_2^2-V_1^2\right)}{Q_+ - k_1x_aR - 4(1-(k_1/k_2))\left(V_2^2-V_1^2\right)}\right),$$

where

$$\begin{aligned} R &= \sqrt{k_1^2x_a^2 - 4\left(V_2^2-V_1^2\right)\left(1-(1-(k_1/k_2))^2\right)}, \\ Q_\pm &= \left\{k_1^4x_a^4 + 4\left(V_1^2 + (1-(k_1/k_2))^2\left(2V_2^2-V_1^2\right)\right)\right. \\ &\quad + 16(1-(k_1/k_2))^2\left(V_2^2-V_1^2\right)\left(V_2^2(1-(k_1/k_2))^2 - V_1^2\right) \\ &\quad \left.\mp 8V_1^2k_1x_a(1-(k_1/k_2))R\right\}^{1/2}. \end{aligned} \tag{67}$$

ACKNOWLEDGMENTS

I am indebted to Fons ten Kroode for pleasant discussions and very useful remarks, to Koos de Vos for crucial suggestions concerning the Appendix and to Wim Mulder for carefully reading the manuscript. I also thank Professor Bill Symes for carefully reading the manuscript and making very valuable suggestions for improving its content. Shell International Exploration and Production B.V. is acknowledged for its kind permission to publish this paper.

REFERENCES

1. J. D. Benamou, *J. Comput. Phys.* **128**, 463 (1996).
2. V. Vinje, E. Iversen, and H. Gjoystdal, *Geophysics* **58**, 1157 (1993).
3. E. Rouy and A. Tourin, *SIAM J. Numer. Anal.* **29**, 867 (1992).
4. W. M. Telford, L. P. Geldart, R. E. Sheriff, and D. A. Keys (*Applied Geophysics*, Cambridge Univ. Press, Cambridge, 1976).
5. J. Milnor, *Annals of Math. Studies*, Vol. 51 (Princeton Univ. Press, Princeton, NJ, 1963).
6. I. M. Gelfand and S. V. Fomin, *Calculus of Variations* (Prentice–Hall, Englewood Cliffs, NJ, 1963).