



A level set based Eulerian method for paraxial multivalued traveltimes

Jianliang Qian^{*}, Shingyu Leung

Department of Mathematics, University of California, 405 Hilgard Avenue, 207 Church Street, SE, Los Angeles, CA 90095-1555, USA

Received 25 June 2003; received in revised form 27 December 2003; accepted 29 December 2003
Available online 27 February 2004

Abstract

We apply the level-set methodology to compute multivalued solutions of the paraxial eikonal equation in both isotropic and anisotropic metrics. This paraxial equation is obtained from 2D stationary eikonal equations by using one of the spatial directions as the artificial evolution direction. The advection velocity field used to move level sets is obtained by the method of characteristics; therefore, the motion of level sets is defined in a phase space, and the zero level set yields the location of bicharacteristic strips in the reduced phase space. The multivalued traveltime is obtained from solving another advection equation with a source term. The complexity of the algorithm is $O(N^3 \log N)$ in the worst case and $O(N^3)$ in the average case, where N is the number of the sampling points along one of the spatial directions. Numerical experiments including the well-known Marmousi synthetic model illustrate the accuracy and the efficiency of the Eulerian method.

© 2004 Elsevier Inc. All rights reserved.

1. Introduction

The eikonal equation as a first-order non-linear PDE usually admits more than one weak solution. The concept of viscosity solution developed by Crandall and et al. [10] and others makes use of the maximum principle for the non-linear PDE and picks out a unique and stable solution among many weak solutions; physically, this solution corresponds to the first arrival or the least traveltime if the solution of the eikonal equation has the dimension of time [21]. However, in practice, later arrivals may carry information which is more relevant to applications. In geophysical oil explorations, for example, the first-arrival wavefront may not carry the most energetic part of the wave-field, and later-arrival wavefronts may be more useful for modern high resolution seismic imaging via integral transform in the presence of strong refraction [15,22,25]. In the quantum mechanics, the WKBJ method for the semi-classical limit of the Schrödinger

^{*} Corresponding author. Tel.: +1-310-825-4746; fax: +1-310-206-6673.

E-mail addresses: qian@math.ucla.edu (J. Qian), syleung@math.ucla.edu (S. Leung).

equation needs multivalued phases to construct the asymptotic expansion of the wave field, where the phases are multivalued solutions of eikonal equations [6,23].

Naturally, one may use the method of characteristics, a Lagrangian formulation, to compute multivalued solutions of the eikonal equation. However, it suffers from some typical shortcomings of Lagrangian methods, for instance, non-uniform distribution of the solutions in the physical space. The resolution of this is via dynamic addition and removal of rays (characteristics) as computation proceeds at the cost of complicated data structures and bookkeeping [44]. Therefore, one looks to Eulerian formulations for computing multivalued solutions of the eikonal equation. In this regard, a new field, so-called Eulerian geometrical optics, emerged [2] as many researchers have devoted a lot of efforts to developing efficient Eulerian methods for computing multivalued solutions since the early 90s. As a result, there are many different approaches in the literature: explicit caustic construction method [1,3], slowness matching method [41,42], segment projection method [13], dynamic surface extension method [37,38], kinetic method for multibranch entropy solutions [4,16], higher co-dimension level-set evolution method [6,7,20,26,31], Liouville equations for escape parameters [14], to name just a few.

The level-set method was first used to compute multivalued phases in the high frequency asymptotics for acoustic wave equations in [26], where the multivalued phases are implicitly represented as self-intersecting wavefronts in the physical space. Later it was extended to compute multivalued phases (traveltimes) in the high frequency asymptotics for anisotropic elastic wave equations in [7,31], where the multivalued solutions for a class of steady Hamilton–Jacobi equations were computed and illustrated as self-intersecting wavefronts as well. Recently, it was extended to compute multivalued wavefronts and multivalued phases in the high frequency asymptotics for the Schrödinger equation in [6], where multivalued solutions for time-dependent Hamilton–Jacobi equations were constructed in a general level-set framework; it was also extended to compute the multivalued gradient of the solution for a class of Hamilton–Jacobi equations in [20], where a level-set formulation for handling the gradient of the solution is used to obtain the Liouville equation, but the formulation only yields the multivalued gradient of the solution and does not provide the multivalued solution itself.

In this paper, we propose another level-set based Eulerian method for computing multivalued solutions of the paraxial eikonal equation in both isotropic and anisotropic metrics. The level-set framework provides a natural link from a Lagrangian formulation to an Eulerian formulation. We first derive the ray tracing equation using one of the spatial directions as the running parameter, which corresponds to the paraxial eikonal equation. The ray tracing equation is embedded into a level-set motion equation to define a passive motion for level sets. The multivalued traveltime is obtained from solving another advection equation with a non-homogeneous source term. The complexity of the algorithm is $O(N^3 \log N)$ in the worst case and $O(N^3)$ in the average case where N is the number of the sampling points along one of the spatial directions. After we finished this paper, we became aware of that our method shares some similarities with that proposed in [6]; but our setup, derivation and aimed applications are different from those presented in [6], and one of the advantages of our work is that multivalued solutions of 2D steady eikonal equations are obtained by solving two “time” dependent (z -dependent), 2D in a reduced phase space $((x, \theta)$ space with θ being the phase angle), linear advection equations rather than two time dependent (t -dependent), 3D in a reduced phase space $((x, z, \theta)$ space), linear advection equations as in [26,31]. Our method is different from that in [20] because our formulation yields both locations of multivalued wavefronts and multivalued traveltimes, while the formulation in [20] only yields the multivalued gradient of the solution. Our method is also different from that in [14], in terms of how to obtain a well-posed problem. Under a point source condition considered here for the eikonal equation, our method is based on an advection equation for traveltimes in phase space $((x, z, \theta)$ space), thus allowing for multivaluedness on projection to physical space. Since the traveltime is not defined for all values of θ at every physical location (x, z) under a point source condition, a level-set function defined in the phase space is advected concurrently to indicate the wavefront location. The paraxial assumption is used to guarantee that the planes $z = \text{const.}$ are non-

characteristic for the evolution, so that the z variable can be used to advance the front; this overall gives a well-posed problem. Because the traveltime at a specific location (x, z) may not be defined for all values of θ , to obtain a well-posed problem the work in [14] imposes point source conditions everywhere on the space boundary; this in essence boils down to making the traveltime defined everywhere in the phase space; finally, to extract multivalued traveltimes at a specific physical location a set of Liouville equations for so-called escape parameters have to be solved as well.

Although the paraxial assumption in the method does not allow overturning rays in the physical space, it is sufficient for many geophysical applications [17,33,34]. We notice that the paraxial formulation for eikonal equations [17] is different from the paraxial (or parabolic) formulation for wave equations [8]. The paraxial formulation for wave equations first appeared in reflection seismology [8], and the resulting equation is usually called the parabolic wave equation; later the idea was used in underwater acoustics and many other applications. One may apply the high frequency asymptotics to a parabolic wave equation to obtain a “time”-dependent eikonal equation as well. However, as pointed out in [12], the paraxial approximation in geometrical optics signifies another simplification which can be made when there is one preferred wave propagation direction so that a stationary eikonal equation can be rewritten as an evolution equation in one of the spatial variables [17,35]; as long as a version of the so-called sub-horizontal condition holds, there is no approximation involved. The paraxial eikonal equation we will use first appeared in [17] and was rigorously justified in [42].

The rest of the paper is organized as follows: Sections 2 and 3 present the paraxial formulation for isotropic and anisotropic eikonal equations, respectively; Section 4 presents the level-set formulation for multivalued wavefronts and multivalued traveltimes; Section 5 gives implementation details for the level-set method; Section 6 demonstrates the accuracy of the proposed Eulerian method with extensive numerical examples, including the well-known Marmousi synthetic model; Section 7 concludes the paper.

2. Paraxial formulation for isotropic eikonal equation

Consider the eikonal equation with a point source condition in an isotropic medium which occupies an open, bounded domain $\Omega \subset \mathbf{R}^2$. By isotropy here we mean the wave velocity has no directional dependence. The equation is as follows:

$$|\nabla_{\mathbf{x}} \tau(\mathbf{x}, \mathbf{x}_s)| = \frac{1}{c(\mathbf{x})}, \quad (1)$$

$$\lim_{\mathbf{x} \rightarrow \mathbf{x}_s} \frac{\tau(\mathbf{x}, \mathbf{x}_s)}{|\mathbf{x} - \mathbf{x}_s|} = \frac{1}{c(\mathbf{x}_s)}, \quad \tau \geq 0, \quad (2)$$

where \mathbf{x}_s is the given source point, $c \in C^1(\Omega)$ is the positive velocity and $|\cdot|$ denotes the two-norm in the \mathbf{x} space. Here, $\tau(\mathbf{x}, \mathbf{x}_s)$ denotes the time (“traveltime”) taken by a particle moving at velocity $c(\mathbf{x})$ to travel from the source point \mathbf{x}_s to a target point $\mathbf{x} \in \Omega$. For $\mathbf{x} \neq \mathbf{x}_s$ near \mathbf{x}_s , τ is a differentiable function of both arguments and satisfies the *eikonal equation* (1). However, when \mathbf{x} is sufficiently distant from \mathbf{x}_s and the velocity $c(\mathbf{x})$ is inhomogeneous with spatial position \mathbf{x} , τ is generally a multivalued function of both variables, and cusps and caustics occur with high probabilities [45].

The concept of viscosity solution can be used to extract a globally single valued solution for the eikonal equation [10]; this solution assigns to each point \mathbf{x} the least of the (possibly many) traveltimes from \mathbf{x}_s to \mathbf{x} . Relying on this concept we may use finite difference schemes to compute the least traveltime stably and efficiently [11,28,29,34–36]. Although the viscosity concept and related numerical methods provide a

natural Eulerian framework for geometrical optics, the drawback is that it provides only first-arrivals while some applications may require all arrival times.

To motivate our derivation from a multivalued Lagrangian framework to a multivalued Eulerian framework, we first derive a single-valued Lagrangian framework from the above viscosity-solution-based single-valued Eulerian framework. Because the rays of geometric optics are the orthogonal trajectories of the wavefronts (level sets of τ) [9], consider the one-parameter family of two-dimensional wavefronts $\gamma(t)$, where $t \in [\epsilon_0, \infty)$ is time, ϵ_0 is a small number and $\gamma(\epsilon_0)$ is a simple, closed and smooth wavefront. Then near $t = \epsilon_0$, $\gamma(t)$ can be generated by moving $\gamma(\epsilon_0)$ along the normal vector field with speed c depending on position \mathbf{x} . Let $\mathbf{x}(r, t) = (x(r, t), z(r, t))$ be the position vector which parameterizes $\gamma(t)$ by $r : 0 \leq r \leq R$, $\mathbf{x}(0, t) = \mathbf{x}(R, t)$. Then the equations of motion can be written as

$$x_t = c(x, z) \frac{z_r}{\sqrt{x_r^2 + z_r^2}}, \quad z_t = -c(x, z) \frac{x_r}{\sqrt{x_r^2 + z_r^2}}. \quad (3)$$

Given the mapping from $[0, R] \times [\epsilon_0, \infty)$ to \mathbf{R}^2 generated by the moving curve, there exists near $t = \epsilon_0$ an inverse mapping function τ defined by $t = \tau(x, y)$. Then the function τ satisfies the eikonal equation (1), as long as the wavefront stays smooth and non-intersecting [28]. If the wavefront intersects, then the particle tracking system (3) without regridding is linearly ill-posed [28]. However, the monotone numerical method for the viscosity solution of the eikonal equation (1) was well developed which yields a physically relevant solution even if the wavefront self-intersects. In fact, the success of the level-set method in the early stage was more or less attributed to the concept of viscosity solution and related high-order numerical methods [19,29], so that topology change and merging can be taken care of automatically when they occur. On the other hand, it also implies that in the current framework, i.e., Lagrangian formulation (3) and Eulerian formulation (1), it is hard to capture self-intersecting wavefronts unless special care is taken to keep track of some extra parameters, such as amplitude [1] or slowness [41,42], etc. These extra parameters are essentially used to parameterize the self-intersection, i.e., multivaluedness, and this viewpoint naturally leads one to consider phase-space formulations for computing multivalued solutions.

By the method of characteristics for the eikonal equation (1) with the point source condition (2), we have a ray tracing system

$$\frac{dx}{dt} = c \sin \theta, \quad (4)$$

$$\frac{dz}{dt} = c \cos \theta, \quad (5)$$

$$\frac{d\theta}{dt} = \sin \theta \frac{\partial c}{\partial z} - \cos \theta \frac{\partial c}{\partial x}, \quad (6)$$

with initial conditions

$$x|_{t=0} = x_s, \quad (7)$$

$$z|_{t=0} = z_s, \quad (8)$$

$$\theta|_{t=0} = \theta_s, \quad (9)$$

where $\mathbf{x} = (x, z)$, $\mathbf{x}_s = (x_s, z_s)$ and θ_s varies from $-\pi$ to π . This is a multivalued Lagrangian formulation because even though the rays in the phase space (x, z, θ) may never intersect, the projected rays in the physical space (x, z) may intersect.

In some applications, for example, wave propagation in reflection seismics [8], the traveltimes of interest are carried by the so-called sub-horizontal rays [17,34,40], where sub-horizontal means “oriented in the positive z -direction”. A convenient characterization for sub-horizontal rays is that

$$\frac{dz}{dt} \geq c \cos \theta_{\max} > 0, \quad (10)$$

for some $0 < \theta_{\max} < \pi/2$. This inequality holds for rays making an angle θ with the vertical satisfying $|\theta| \leq \theta_{\max} < \pi/2$.

To be specific, consider

$$\Omega = \{(x, z) : x_{\min} \leq x \leq x_{\max}, 0 \leq z \leq z_{\max}\}, \quad (11)$$

and assume that the source is located on the surface: $x_{\min} \leq x_s \leq x_{\max}$ and $z_s = 0$. By the sub-horizontal condition we can use depth as the running parameter so that we have a reduced system

$$\frac{dx}{dz} = \tan \theta, \quad (12)$$

$$\frac{d\theta}{dz} = \frac{1}{c} \left(\frac{\partial c}{\partial z} \tan \theta - \frac{\partial c}{\partial x} \right), \quad (13)$$

with

$$x|_{z=0} = x_s, \quad (14)$$

$$\theta|_{z=0} = \theta_s, \quad (15)$$

where now θ_s varies from $-\theta_{\max} \leq \theta \leq \theta_{\max} < \pi/2$. In addition, the traveltime is computed by integrating

$$\frac{dt}{dz} = \frac{1}{c \cos \theta}, \quad (16)$$

with

$$t|_{z=0} = 0. \quad (17)$$

This ray tracing system (12)–(15) is a multivalued Lagrangian formulation defined in the reduced phase space $(z; x, \theta)$. Actually, the ray tracing system can be obtained by applying the method of characteristics to the paraxial eikonal equation

$$\frac{\partial \tau}{\partial z} = H\left(x, z, \frac{\partial \tau}{\partial x}\right) = \sqrt{\max\left(\frac{1}{c^2} - \left(\frac{\partial \tau}{\partial x}\right)^2, \frac{\cos^2 \theta_{\max}}{c^2}\right)}, \quad (18)$$

which in turn comes from enforcing the sub-horizontal condition in the eikonal equation (1); see [42] for a theoretical justification.

As we will see, since the ray tracing system (12), (13) is formulated in a reduced phase space, we may use a two-dimensional level-set motion equation to move the initial curve deduced from the initial condition and the curves moved will not self-intersect because they are defined in the reduced phase space.

3. Paraxial formulation for anisotropic eikonal equation

For general anisotropic media in which wave propagation velocities have both spatial and directional dependence, we may also formulate paraxial eikonal equations by enforcing a version of the sub-horizontal condition.

To illustrate the idea behind our approach, we consider the two-dimensional anisotropic eikonal equation only; please see [31] for a detailed derivation of 3D anisotropic eikonal equations. Consequently, we denote the 2D anisotropic eikonal equation as

$$F(x, z, p_1, p_3) = 0, \quad (19)$$

where F is a function depending on the anisotropic medium under consideration. Here, $p_1 = \tau_x$ and $p_3 = \tau_z$ are components of the slowness vector $\nabla\tau$ with τ being the travelt ime. Parameterize the slowness vector by

$$p_1 = \frac{\sin \theta}{V(x, z, \theta)}, \quad p_3 = \frac{\cos \theta}{V(x, z, \theta)}, \quad (20)$$

where θ is known as the phase angle, varying from $-\pi$ to π , and V as the phase velocity solving an eigenvalue problem [31] and selecting different wave modes. Applying the method of characteristics to Eq. (19) yields

$$\frac{dx}{dt} = \left(p_1 \frac{\partial F}{\partial p_1} + p_3 \frac{\partial F}{\partial p_3} \right)^{-1} \frac{\partial F}{\partial p_1}, \quad (21)$$

$$\frac{dz}{dt} = \left(p_1 \frac{\partial F}{\partial p_1} + p_3 \frac{\partial F}{\partial p_3} \right)^{-1} \frac{\partial F}{\partial p_3}, \quad (22)$$

$$\frac{dp_1}{dt} = - \left(p_1 \frac{\partial F}{\partial p_1} + p_3 \frac{\partial F}{\partial p_3} \right)^{-1} \frac{\partial F}{\partial x}, \quad (23)$$

$$\frac{dp_3}{dt} = - \left(p_1 \frac{\partial F}{\partial p_1} + p_3 \frac{\partial F}{\partial p_3} \right)^{-1} \frac{\partial F}{\partial z}, \quad (24)$$

where the normalization is made so that the evolution parameter t has the dimension of time and thus it is identical to τ . To obtain an equation for $d\theta/dt$, we differentiate the equations in (20), arriving at

$$\frac{dp_1}{dt} = \frac{V \cos \theta - \frac{\partial V}{\partial \theta} \sin \theta}{V^2} \frac{d\theta}{dt} - \frac{\sin \theta}{V^2} \left(\frac{\partial V}{\partial x} \frac{dx}{dt} + \frac{\partial V}{\partial z} \frac{dz}{dt} \right), \quad (25)$$

$$\frac{dp_3}{dt} = \frac{-V \sin \theta - \frac{\partial V}{\partial \theta} \cos \theta}{V^2} \frac{d\theta}{dt} - \frac{\cos \theta}{V^2} \left(\frac{\partial V}{\partial x} \frac{dx}{dt} + \frac{\partial V}{\partial z} \frac{dz}{dt} \right). \quad (26)$$

Thus, solving the above equations for $d\theta/dt$ and substituting in (23) and (24) gives us

$$\frac{d\theta}{dt} = \left(p_1 \frac{\partial F}{\partial p_1} + p_3 \frac{\partial F}{\partial p_3} \right)^{-1} \left(V \frac{\partial F}{\partial x} \cos \theta - V \frac{\partial F}{\partial z} \sin \theta \right). \quad (27)$$

Eqs. (21), (22) and (27) give us the ray tracing system which may be solved with suitable initial conditions as (7)–(9).

The condition for sub-horizontal rays is

$$\frac{dz}{dt} = \left(p_1 \frac{\partial F}{\partial p_1} + p_3 \frac{\partial F}{\partial p_3} \right)^{-1} \frac{\partial F}{\partial p_3} > 0, \quad (28)$$

which may be enforced easily in the reduced phase space for different wave modes. This implies that we may use the depth variable as the running parameter along the ray so that we have

$$\frac{dx}{dz} = \frac{\partial F}{\partial p_1} \left(\frac{\partial F}{\partial p_3} \right)^{-1}, \quad (29)$$

$$\frac{d\theta}{dz} = \left(\frac{\partial F}{\partial p_3} \right)^{-1} \left(V \frac{\partial F}{\partial x} \cos \theta - V \frac{\partial F}{\partial z} \sin \theta \right), \quad (30)$$

augmented with initial conditions (14) and (15).

Similar to the isotropic case, the traveltime $T = t$ is computed by integrating

$$\frac{dT}{dz} = p_3 + p_1 \frac{dx}{dz}, \quad (31)$$

with the initial condition (17).

4. Level set formulation

As we mentioned above, we treat z as an artificial time variable. Now, if we define $\phi = \phi(z, x, \theta)$ such that the zero level set, $\{(x(z), \theta(z)) : \phi(z, x(z), \theta(z)) = 0\}$, gives the location of the reduced bicharacteristic strip $(x(z), \theta(z))$ at z , then we may differentiate the zero level-set equation with respect to z to obtain

$$\phi_z + u\phi_x + v\phi_\theta = 0, \quad (32)$$

with

$$u = \frac{dx}{dz} \quad \text{and} \quad v = \frac{d\theta}{dz}, \quad (33)$$

which are given by the ray equations (12) and (13) or (29) and (30). In essence, we embed the ray tracing equations as the velocity field, $\mathbf{u} = (u, v)$, into the level set equation which governs the motion of the bi-characteristic strips in the phase space.

The initial condition for the level-set motion equation (32) is taken to be

$$\phi|_{z=0} = \phi(0, x, \theta) = x - x_s, \quad (34)$$

which is obtained from initial conditions (14) and (15). This is a signed distance function, satisfying $|\nabla_{x,\theta}\phi| = 1$, to the initial phase space curve

$$\{(x, \theta) : x = x_s, -\theta_{\max} \leq \theta \leq \theta_{\max}\}, \quad (35)$$

in the reduced phase space

$$\Omega_p = \{(x, \theta) : x_{\min} \leq x \leq x_{\max}, -\theta_{\max} \leq \theta \leq \theta_{\max}\}. \quad (36)$$

The initial curve partitions Ω_p into two sub-domains represented by $\{(x, \theta) : \phi(0, \cdot, \cdot) < 0\}$ and $\{(x, \theta) : \phi(0, \cdot, \cdot) > 0\}$. Afterwards, the level-set motion equation takes over and moves this initial curve as z varies, and the zero level set of ϕ at z gives the location of the new curve which still partitions Ω_p into two sub-domains. Since the initial curve defines an implicit function between x and θ , where θ is a (possibly multivalued) function of x , the new curve shares the same property. Therefore, for fixed z , for some x s we may have more than one θ^* such that $\phi(z, x, \theta^*) = 0$. This essentially tells us where the solutions are multivalued.

To determine the arrival-time of the ray from the above level-set equation, we now derive a corresponding equation governing the evolution of the traveltime. By the sub-horizontal condition in the paraxial formulation and the ray equation (16) or (31), let $F_u(z; x, \theta)$ be the flow generated by the velocity field $\mathbf{u} = (u, v)$ in the phase space (x, θ) along the z -direction. Then we can write

$$\frac{dT}{dz}(z, F_u(z; x, \theta)) = \frac{1}{c \cos \theta}, \quad (37)$$

in the isotropic case and

$$\frac{dT}{dz}(z, F_u(z; x, \theta)) = p_3 + p_1 \frac{dx}{dz}, \quad (38)$$

in the anisotropic case. Therefore, having $t = T(z, x, \theta)$, we get the following advection equation:

$$\frac{dt}{dz} = \frac{dT}{dz} = T_z + uT_x + vT_\theta = \frac{1}{c \cos \theta}, \quad (39)$$

for isotropic traveltime and

$$\frac{dt}{dz} = \frac{dT}{dz} = T_z + uT_x + vT_\theta = p_3 + p_1 \frac{dx}{dz}, \quad (40)$$

for anisotropic traveltime.

The initial condition for T is specified according to the initial condition (17)

$$T|_{z=0} = T(0, x, \theta) = 0, \quad (41)$$

which is consistent with the initial condition (34).

By solving the level-set equation (32), we have the locations of wavefronts; by solving the traveltime equation (39) or (40), we have the traveltime values at the corresponding locations. Therefore, the multivalued traveltime at a specific location in the physical space can be obtained by first computing wavefront locations and then interpolating the traveltime at that location from gridded traveltimes.

5. Implementation

We will give full details on implementing the level-set Eulerian method for isotropic eikonal equations only.

5.1. Boundary conditions and an algorithm

Because the level-set equation is a homogeneous advection equation, a non-reflective boundary condition is used for the level-set equation, meaning $\partial\phi/\partial\mathbf{n} = 0$, where \mathbf{n} is the outward normal along the

boundary of Ω_p . This ensures the information outside the domain Ω_p will not interfere with the zero level set inside the computational domain.

However, since the traveltime equation is a non-homogeneous advection equation, we need to specify normal derivatives of the traveltime along the boundary according to the characteristic system. In general, there are two types of characteristics, i.e., outgoing and incoming characteristics. By using upwind numerical schemes, those boundary conditions can be taken care of automatically if some appropriate normal derivatives are imposed on the boundary so that the outside information will not propagate into the computational domain. To obtain such normal derivatives, we use the information from the characteristics system. We first invert Eqs. (4) and (6) locally and get

$$\frac{\partial T}{\partial x} = \frac{1}{c \sin \theta}, \quad (42)$$

which is used for boundaries $x = x_{\min}$ and $x = x_{\max}$, and

$$\frac{\partial T}{\partial \theta} = \left(\sin \theta \frac{\partial c}{\partial z} - \cos \theta \frac{\partial c}{\partial x} \right)^{-1}, \quad (43)$$

which is for boundaries $\theta = -\theta_{\max}$ and $\theta = \theta_{\max}$.

The above conditions essentially specify normal derivatives of the traveltime along the boundaries. Then the values of T on the boundaries will be obtained by applying the Adams' Extrapolation formula to Eq. (42), where θ is considered as fixed, and to Eq. (43), where x is considered as fixed.

In Eq. (42), when $\theta = 0$, it seems that there is a singularity. However, in that case, $u = \tan \theta = 0$, and thus the information on the boundary will not propagate into the computational domain.

With above ingredients in place, we summarize the first algorithm for determining the multivalued traveltimes for all x at some depth z^* .

Algorithm 1.

- (I) Solve the level-set equation (32) and the traveltime equation (39) up to z^* with the velocity field generated by the ray equations (12) and (13).
- (II) For all x ,
 - (i) determine **all** θ_i such that $\phi(z^*, x, \theta_i) = 0$ ($i = 1, \dots$) by root finding;
 - (ii) determine $T(z^*, x, \theta_i)$ ($i = 1, \dots$) by interpolation.

In Step I, the level-set equation and the traveltime equation are decoupled and can be solved separately. The x and θ derivatives are approximated by a fifth-order WENO-Godunov scheme [19] while a third-order TVD-RK method [29] can be used for the z derivative. Since both the level-set equation (32) and the traveltime Eq. (39) are linear, the CFL step Δz can be chosen by

$$\Delta z \leq C \frac{\min(\Delta x, \Delta \theta)}{\max(\sqrt{u^2 + v^2})}, \quad (44)$$

where Δx and $\Delta \theta$ are mesh sizes along x - and θ -directions, respectively, and C is a CFL number taken to be 0.6. For the root finding and the interpolation in Step II, we can simply use any non-oscillatory interpolation scheme, for example, linear interpolation or ENO reconstruction. For z^* fixed the root finding on each line of x is conducted by checking the values of ϕ with θ varying. Interpolation for traveltime is performed where a root is present according to the Intermediate Value Theorem.

5.2. Regularizations

Initially at $z = 0$, we have a signed distance function satisfying $|\nabla\phi| = 1$, so that the level sets, i.e., contours, of ϕ are equally spaced. However, as z varies the level-set equation is solved and the level-set function is updated; in general the level-set function is no longer equally spaced because of the underlying inhomogeneous velocity field, even though the zero level set of ϕ at z , the curve that we are interested in, is moving at the correct velocity. This implies that ϕ may develop steep and flat gradients at or near the zero level set, making the computed curve locations and further computations inaccurate, which does happen in Algorithm 1.

In fact, numerically, if Algorithm 1 is implemented straightforwardly, then the obtained multivalued solution is inaccurate. Therefore, we propose the following regularization procedure which consists of reinitialization and orthogonalization.

To restore the equally spaced property for the level sets, the usual way is to make ϕ a signed distance function without moving the zero level set of ϕ appreciably. This can be achieved through so-called reinitialization and the usual way is to solve the following equation to steady state $\tilde{\phi}_\infty$ [18,27,30,39]:

$$\frac{\partial \tilde{\phi}}{\partial \xi} + S(\phi) \left(|\nabla \tilde{\phi}| - 1 \right) = 0, \quad (45)$$

$$\tilde{\phi}|_{\xi=0} = \phi(z, \cdot, \cdot), \quad (46)$$

$$\left. \frac{\partial \tilde{\phi}}{\partial \mathbf{n}} \right|_{\partial \Omega_p} = 0, \quad (47)$$

where $S(\phi)$ is a smooth, bounded, strictly monotone increasing function of ϕ with $S(0) = 0$ [18]. In practice, we have found that the following choice of $S(\phi)$ works well in the numerical examples presented below:

$$S(\phi) = \frac{2}{\pi} \tan^{-1}(\phi). \quad (48)$$

There are other possible choices, such as mesh-dependent smeared signum functions; see [27] for more examples.

The steady state $\tilde{\phi}_\infty$ has the same zero level set as $\phi(z, \cdot, \cdot)$ within a certain accuracy since $\tilde{\phi}$ does not move on the zero level set of ϕ . Moreover, at the steady state $\tilde{\phi}_\infty$ is a signed distance function since $|\nabla \tilde{\phi}_\infty| = 1$. The reinitialization step is to use $\tilde{\phi}_\infty$ instead of $\phi(z, \cdot, \cdot)$ as the initial condition at z for solving the level-set motion equation to the next stage. To achieve the steady state, we usually need only evolve Eq. (45) for a few steps of the reinitialization procedure. How often we should invoke the reinitialization procedure is a subtle issue; see [27,30] for some discussions. In our implementation, we invoke the reinitialization at every z step so that we have a better-behaved function for determining the values θ_i in Step II of Algorithm 1.

Even with careful implementation of the above reinitialization procedure, the location of the zero level set may still be shifted by an amount less than one grid cell. This is harmless for the visualization purpose of the location of the wavefront. However, because the solution from the traveltime equation (39) would typically vary a lot near the corresponding location of the zero level set of ϕ , this shift makes the results from the interpolation in Step II highly inaccurate.

Because we are only interested in the value of T where $\phi = 0$, we propose the following orthogonalization procedure:

$$\frac{\partial \tilde{T}}{\partial \xi} + S(\phi) \left(\frac{\nabla \phi}{|\nabla \phi|} \cdot \nabla \tilde{T} \right) = 0, \quad (49)$$

$$\tilde{T}|_{\xi=0} = T(z, \cdot, \cdot), \quad (50)$$

$$\left. \frac{\partial \tilde{T}}{\partial \mathbf{n}} \right|_{\partial \Omega_p} = 0, \quad (51)$$

which, theoretically, preserves the values of T where $\phi = 0$ but changes them elsewhere such that the new T would not vary too much near the desired region. At the steady state, $\nabla \phi \cdot \nabla \tilde{T} = 0$. Eq. (49) may also be viewed as an extension procedure; namely, we extend the values of T on the zero level set of ϕ along the normals of the zero level set of ϕ ; see [27,30] and also see [5] for orthogonalization in higher co-dimensional level-set methods. This generally makes T discontinuous since lines normal to the zero level set will eventually intersect somewhere away from the zero level set. Even if the location of the zero level set may be shifted, the effect to the interpolation will still be acceptable as long as the computational mesh is fine enough; see Section 6 for more discussions.

Similar to the reinitialization in ϕ , we only need to apply several iterations, instead of solving it until reaching the steady state. This makes the regularization procedure efficient, simple to implement and robust.

Incorporating the regularization procedure into Algorithm 1, we have an improved algorithm.

Algorithm 2.

- (I) Initialization: given N_z , N_x and N_θ : $\Delta z = \frac{z_{\max}}{N_z-1}$, $\Delta x = \frac{x_{\max}-x_{\min}}{N_x-1}$ and $\Delta \theta = \frac{2\theta_{\max}}{N_\theta-1}$; initialize ϕ and T at $z = 0$.
- (II) For $k = 1$ to N_z :
 1. March one Δz step from $(k-1)\Delta z$ to $k\Delta z$ by solving the level-set equation (32) and reinitializing the level-set motion by solving (45) at every intermediate z -step.
 2. March one Δz step from $(k-1)\Delta z$ to $k\Delta z$ by solving the traveltime equation (39).
 3. Orthogonalize T and ϕ by solving Eq. (49).
 4. For $x = (j-1)\Delta x, j = 1, \dots, N_x$,
 - (i) determine **all** θ_i such that $\phi(k\Delta z, x, \theta_i) = 0$ ($i = 1, \dots$) by root finding;
 - (ii) determine $T(k\Delta z, x, \theta_i)$ ($i = 1, \dots$) by interpolation.

Since the reinitialization procedure is usually invoked for a fixed number of steps (from 1 to 2 steps in our numerical examples presented below), the above algorithm in the average case has the complexity $O(N^3)$ where $N_z = N_x = N_\theta = N$ is assumed. In the worst case, if the reinitialization procedure is invoked until convergence, the above algorithm has the complexity $O(N^3 \log N)$.

6. Numerical experiments

For the first three examples, we put a point source at the origin and the velocity functions $c(x, z)$ are all C^∞ . The fourth example, the synthetic Marmousi model, is a more challenging one where the velocity function is given only as a sampled function.

In all the examples the computational domain is chosen to be

$$\Omega_p = \{(x, \theta) : -1 \leq x \leq 1, -\theta_{\max} \leq \theta \leq \theta_{\max}\}, \quad (52)$$

with $\theta_{\max} = 9\pi/20$. Accordingly, the Marmousi velocity model will be rescaled to the above computational domain. The last example is an anisotropic model which consists of three different wave modes; one of the wave modes has the so-called instantaneous singularity, and the corresponding traveltimes are multivalued.

6.1. Constant velocity model

When the velocity c is constant, the analytic solution for the traveltimes is known so that we can study the accuracy and the convergence order of the proposed Eulerian method. We compute the traveltimes up to $z = 1.0$ km with different options of regularization procedures to see how reinitialization and orthogonalization affect the accuracy and the convergence order of the method.

Fig. 1 shows the evolution of the zero level set as the depth z increases. Notice that initially we have a vertical line. Under the influence of the velocity field $\mathbf{u} = (\tan \theta, 0)$, the upper part of the vertical line is advected to the right, and the lower part of the vertical line is advected to the left. However, since the

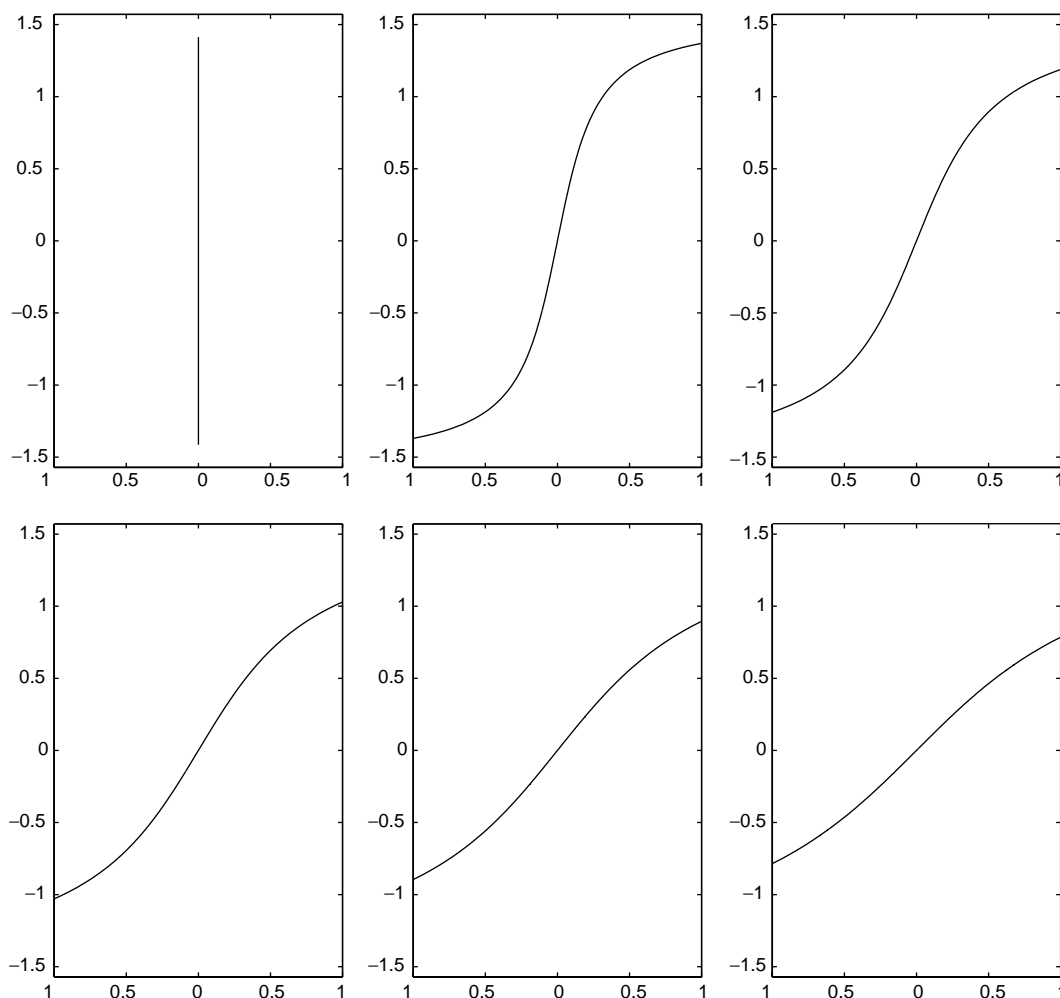


Fig. 1. Evolution of the zero level set in the phase space at $z = 0.0, 0.2, 0.4, 0.6, 0.8$ and 1.0 , respectively.

traveltime is single valued in the constant velocity case, the zero level set always defines a single valued implicit function between x and θ for every fixed z .

Table 1 shows clean second-order accuracy and convergence of traveltimes in l_1 , l_2 and l_∞ norms without either reinitialization or orthogonalization. This is expected because the linear interpolation is used to find roots and extract traveltimes, yielding second-order accuracy only, even though the level-set equation and the traveltime equation are solved to third-order accuracy.

Table 2 also shows second-order convergence of traveltimes in different norms without orthogonalization but with two steps of reinitialization procedure at each z -step. This shows that the reinitialization procedure alone does not move the zero level set too much so that the accuracy is not affected appreciably.

Table 3 shows second-order convergence of traveltimes in different norms without reinitialization but with two steps of orthogonalization procedure at each z -step. This shows that the orthogonalization procedure alone does not change too much the behavior of the traveltime field near the zero level set so that the traveltime accuracy is not affected appreciably.

Table 4 shows neat second-order convergence of traveltimes in different norms when both reinitialization and orthogonalization are invoked. This indicates that combining reinitialization and orthogonalization procedures together does enhance the algorithmic behavior and improve the accuracy of computed traveltimes significantly.

Table 1

Accuracy and convergence order of traveltimes without either reinitialization or orthogonalization

Δx	l_1 error	l_1 order	l_2 error	l_2 order	l_∞ error	l_∞ order
0.20000	0.01374575		0.01153032		0.01213886	
0.10000	0.00366580	1.9067	0.00283586	2.0235	0.00265964	2.1903
0.05000	0.00092272	1.9901	0.00073505	1.9478	0.00079357	1.7447
0.02500	0.00024266	1.9269	0.00018498	1.9904	0.00021538	1.8814
0.01250	0.00005863	2.0491	0.00004600	2.0074	0.00005231	2.0416
0.00625	0.00001497	1.9692	0.00001165	1.9814	0.00001369	1.9336

Table 2

Accuracy and convergence order of traveltimes without orthogonalization but two reinitialization pseudo steps at each z -step using the approximation (48)

Δx	l_1 error	l_1 order	l_2 error	l_2 order	l_∞ error	l_∞ order
0.20000	0.02029779		0.01874370		0.02519635	
0.10000	0.00532929	1.9293	0.00430244	2.1231	0.00484257	2.3793
0.05000	0.00135341	1.9773	0.00116012	1.8908	0.00173133	1.4838
0.02500	0.00035766	1.9199	0.00029215	1.9894	0.00050259	1.7844
0.01250	0.00008633	2.0506	0.00007299	2.0008	0.00011961	2.0710
0.00625	0.00002233	1.9506	0.00001873	1.9617	0.00003397	1.8159

Table 3

Accuracy and convergence order of traveltimes without reinitialization but two orthogonalization pseudo steps at each z -step using the approximation (48)

Δx	l_1 error	l_1 order	l_2 error	l_2 order	l_∞ error	l_∞ order
0.20000	0.00727231		0.00770262		0.01124069	
0.10000	0.00143178	2.3445	0.00131120	2.5544	0.00181632	2.6296
0.05000	0.00040647	1.8165	0.00038993	1.7495	0.00065443	1.4726
0.02500	0.00010576	1.9422	0.00009684	2.0095	0.00020036	1.7076
0.01250	0.00002598	2.0252	0.00002437	1.9899	0.00004782	2.0667
0.00625	0.00000671	1.9517	0.00000633	1.9446	0.00001484	1.6880

Table 4

Accuracy and convergence order of traveltimes with two reinitialization pseudo steps and two orthogonalization pseudo steps at each z -step using the approximation (48)

Δx	l_1 error	l_1 order	l_2 error	l_2 order	l_∞ error	l_∞ order
0.20000	0.00393867		0.00383178		0.00440058	
0.10000	0.00109580	1.8457	0.00097230	1.9785	0.00122641	1.8432
0.05000	0.00027266	2.0067	0.00024258	2.0029	0.00032516	1.9152
0.02500	0.00007131	1.9349	0.00006096	1.9924	0.00008355	1.9603
0.01250	0.00001745	2.0304	0.00001520	2.0031	0.00002115	1.9814
0.00625	0.00000434	2.0053	0.00000380	1.9996	0.00000530	1.9955

6.2. Wave guide

The velocity function is

$$c(x, z) = 1.1 - \exp(-0.5x^2). \quad (53)$$

The function is symmetric with respect to $x = 0$, and we also expect the same type of symmetry in the traveltime.

Fig. 2 shows the computed traveltimes at different depth z s by using only 40×40 grids in the x – θ space. The results shown are the traveltimes at z being 0.8, 1.2, 1.6 and 2.0, respectively. Here, the steps in the reinitialization and orthogonalization procedures are taken to be 1. The solutions are symmetric as expected, and the developing of multivalued traveltime branches is clearly seen.

Fig. 3 shows the zero level set overlaying the traveltime field at $z = 0.0, 0.4, 0.8, 1.2, 1.6$ and 2.0 , respectively. The dashed line is the location of the zero level set and the solid lines are the contour plot of the traveltime function T . There are discontinuities in the traveltime field coming from the orthogonalization procedure, where the normals of the zero level-set intersect. However, since we only use the information near the dashed line, the jump in T will not interfere with our interpolation procedure if the grids are fine enough, meaning that the discontinuity is at least one Δx distance away from the zero level set. Theoretically, to resolve the zero level set itself on a given mesh, two parallel level-set segments should be greater than one Δx distance away from each other so that they can be resolved. In our computation here, we need to perform the orthogonalization procedure, and the normals from the two parallel level set segments intersect in the middle so that discontinuities appear in the traveltime field. To avoid discontinuities interfering with our computation, we have to keep discontinuities away from the zero level set at least one Δx away. This in turn requires that the computational mesh resolve the parallel segments of the zero level set greater than $2\Delta x$ distance away. At the tip of the zero level set, the traveltime is continuous along the zero level set, so discontinuities in the traveltime field will not hurt the computation. Computationally, due to the smearing of discontinuities, the required mesh may be finer.

Fig. 3 also shows that the contours are perpendicular to the zero level set as designed. As z varies, the zero level set is advected so that it has more turnarounds and it no longer defines a single-valued implicit function between x and θ . This in turn implies that the traveltime becomes multivalued, and the number of traveltime arrivals increases from 1 to 3 during this process.

6.3. Sinusoidal model

This example is adapted from the sinusoidal waveguide model proposed in [41,42], and the velocity function is given by

$$c(x, z) = 1 + 0.2 \sin(0.5\pi z) \sin[3\pi(x + 0.55)]. \quad (54)$$

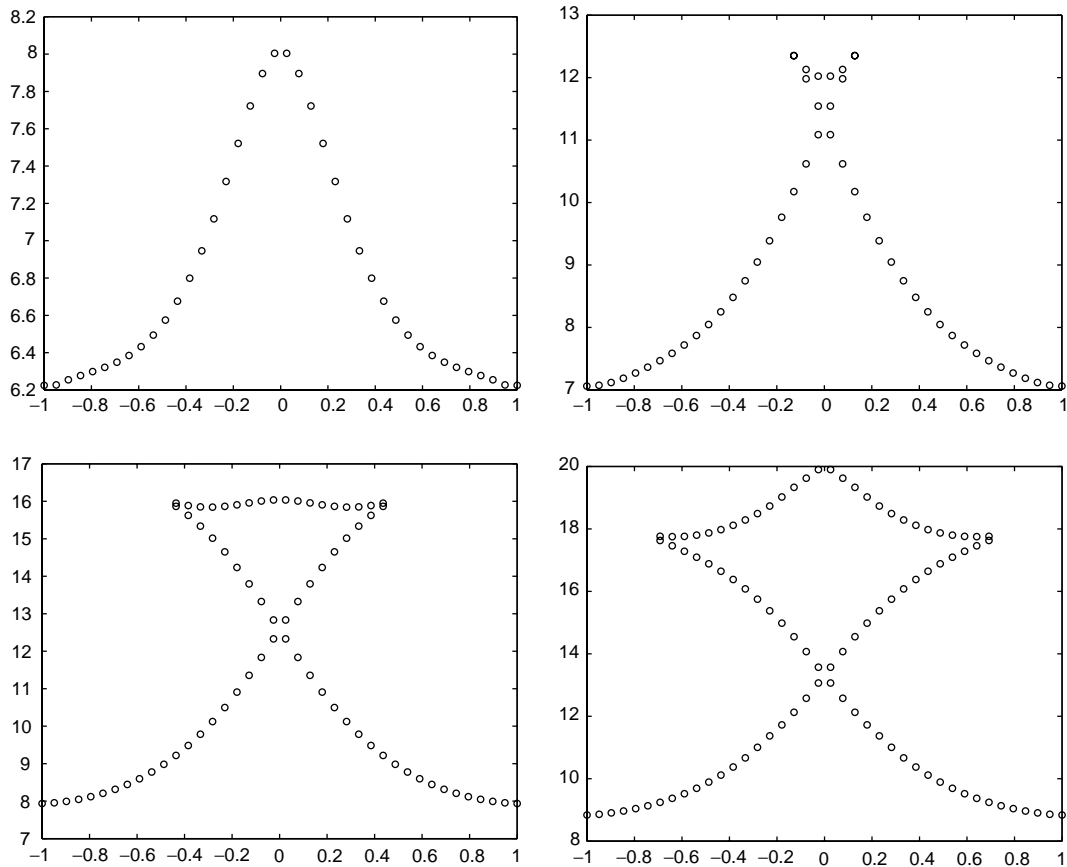


Fig. 2. Multivalued traveltimes by the level-set method for $z = 0.8, 1.2$ (in the upper row), 1.6 and 2.0 in the wave guide model. Multivalued solutions develop as z increases.

Computational results using a 120 by 120 grid in x – θ space are given in Figs. 4–7. In these computations the number of steps in the reinitialization and orthogonalization procedures is set to be 2, respectively. Fig. 4 shows that the triplications in the traveltime developed at $z = 0.4, 1.0, 1.2$ and 1.8 are clearly captured by the level set Eulerian method. Fig. 5 shows the evolution of the zero level set in the phase space. The zero level set is overturned by the underlying velocity field, and this introduces multivaluedness in the traveltime. Fig. 6 shows the zero level set overlaying the traveltime field at $z = 1.2$ and 2.0. Note that there are five traveltimes at some locations x .

To check the accuracy of the computed multivalued traveltimes, we compare the results with the ray tracing solution at $z = 2.0$; the comparison is shown in Fig. 7. The solid line is the solution using our level-set formulation while the circles represent the ray-tracing solution. The solutions match with each other; however, the ray tracing Lagrangian method failed to assign traveltimes to some locations, leaving some shadow zones in the domain, but the level-set Eulerian method here has no such problem at all, and every location was assigned at least one traveltime, i.e., the first-arrival traveltime.

To study the convergence behavior of the method, we also compute the solution up to $z = 2.0$ km using a 240 by 240 grid in x – θ space; Fig. 8 shows the results. In comparison to Fig. 7, the traveltime accuracy in Fig. 8 is improved considerably.

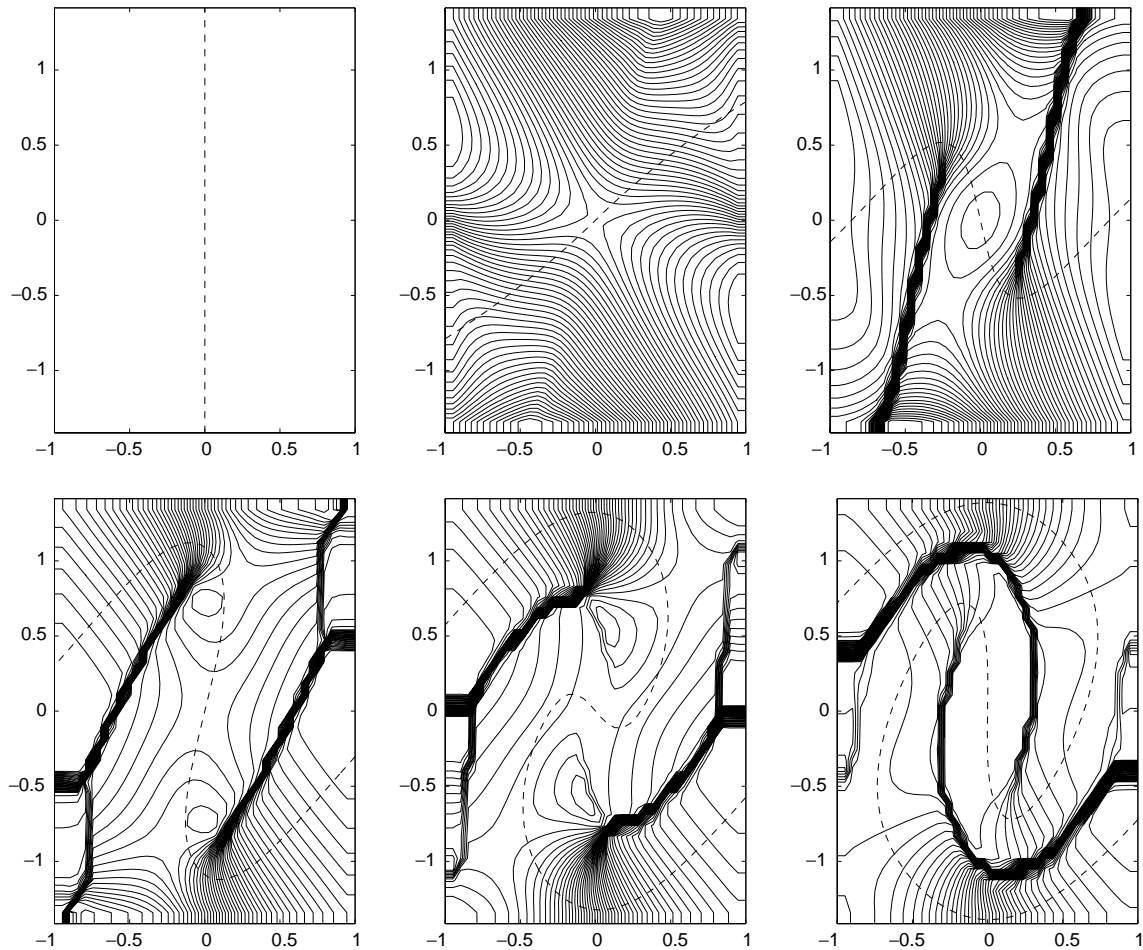


Fig. 3. The zero level-set overlaid on contours of time field T in the wave guide model at different z s: 0.0, 0.4, 0.8, 1.2, 1.6 and 2.0.

6.4. Synthetic Marmousi model

This example is the Marmousi model from the 1996 INRIA Workshop on Multi-arrival Traveltimes. The calibration data used here were computed by Dr. Klimes and can be found at <http://www.caam.rice.edu/~benamou/traveltimes.html>. This is a synthetic model which will challenge the level-set method used here; thus we will study this model carefully.

The original Marmousi model is sampled on a 24 m by 24 m grid, consisting of 384 samples in the x -direction and 122 samples in the z -direction; therefore, the model dimension is 9.192 km long in the x -direction and 2.904 km deep in the z -direction. In the computational results presented here, we use a portion of Marmousi model, i.e., a window from 4.8 to 7.2 km in the x -direction and from 0 to 2.904 km in the z -direction. The source is located at $x = 6.0$ km and $z = 2.8$ km. The purpose is to compute (possibly multivalued) traveltimes for those sampling points, i.e., the receivers from 200 to 300 on the surface $z = 0.0$ km.

In the computation, we have set both reinitialization and orthogonalization steps to be 2 with $S(\phi)$ being given in (48). To speed up the computation in the reinitialization and orthogonalization procedures, we use a first-order Euler scheme in the depth marching and a third-order WENO in the spatial discretization.

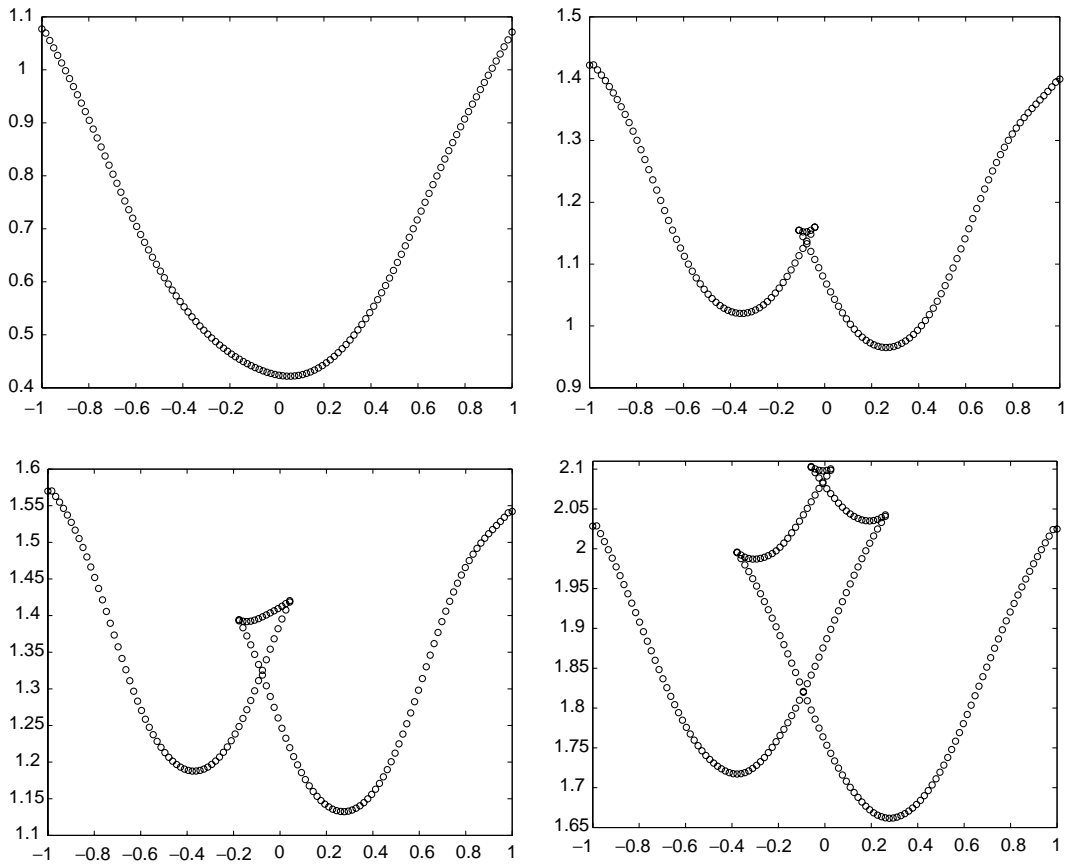


Fig. 4. Traveltimes for $z = 0.4, 1.0$ (in the upper row), 1.2 and 1.8 in the Sinusoidal model. Notice that there are five traveltimes for some points near $x = 0$.

In the first run, we used a 100 by 200 grid in x – θ space with $\Delta x = 24$ m and $\theta_{\max} = 9\pi/20$. The computed traveltime at $z = 0.0$ km and the comparison with the ray tracing solution are shown in Fig. 9. The ray tracing data used to calibrate the computed Eulerian solutions are presumably accurate. As we can see from Fig. 9, the computed Eulerian solution is consistent with the ray tracing solution, being able to capture most of the structure of the multivalued solution, but it failed to resolve some fine details, especially the two traveltime branches located from receivers 260 to 280, where those two branches are very close to each other. Fig. 10 tells us why the level-set method failed in that region. Fig. 10 shows the zero level set and its overlay on the traveltime field in the reduced phase space. From the ray tracing solution we know that receivers from 260 to 280 should have three arrivals, but from the zero level set, receivers from 260 to 280 are single-valued functions of θ , and they correspond to first-arrival traveltimes. Apparently, the tip of the zero level set near receiver 260 should be more elongated, but somehow the level set failed to elongate that tip. This is partly due to the dissipation of the finite difference scheme used here and partly due to the resolution capability of the level-set method which can resolve the zero level set only up to one grid-cell width. Computationally, if the segments of the zero level curve get too close to each other, then they will merge and this is exactly happening to the tip that we are interested in.

Therefore, to resolve the fine tip we have to use a finer grid, 400×200 on x – θ space. Since the original velocity model is given on the discretized points, we use interpolation to obtain a velocity model for the

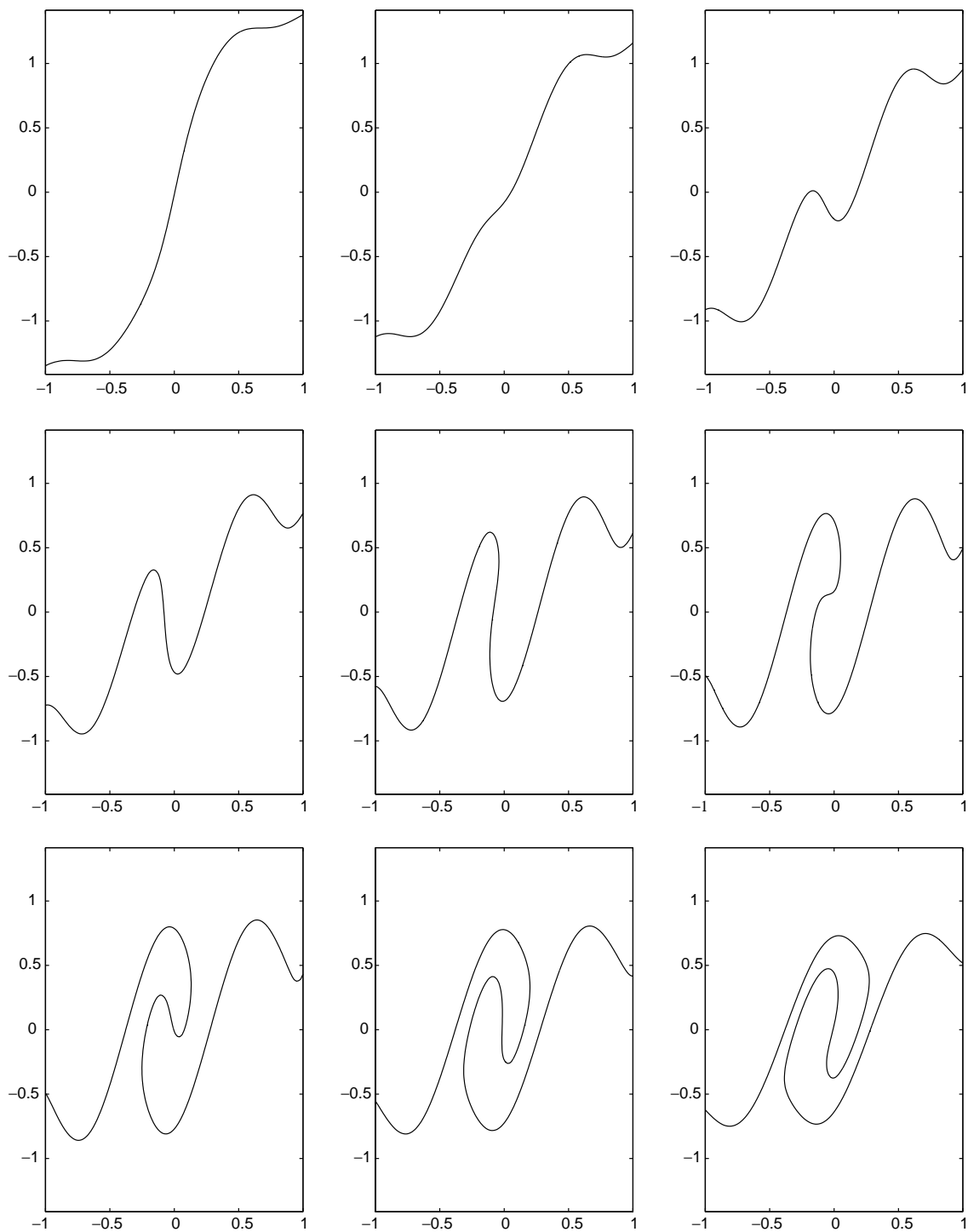


Fig. 5. Evolution of the zero level set in the phase space at different z s from 0.2 to 1.8 with increment of 0.2.

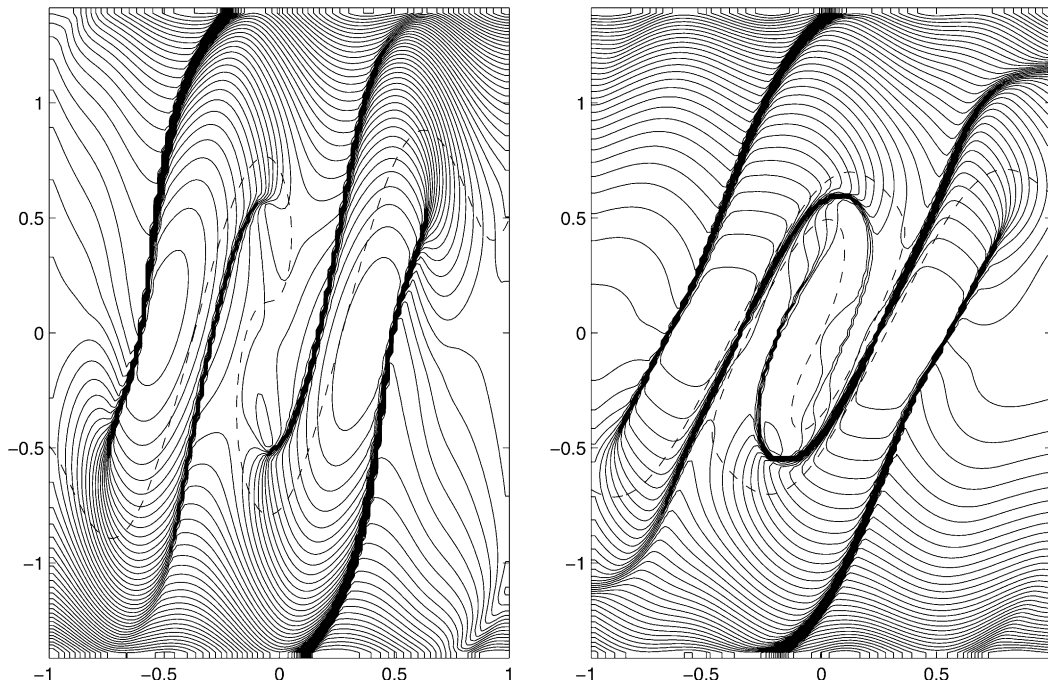


Fig. 6. The zero level set and the contour of T at $z = 1.2$ and 2.0 in Sinusoidal model.

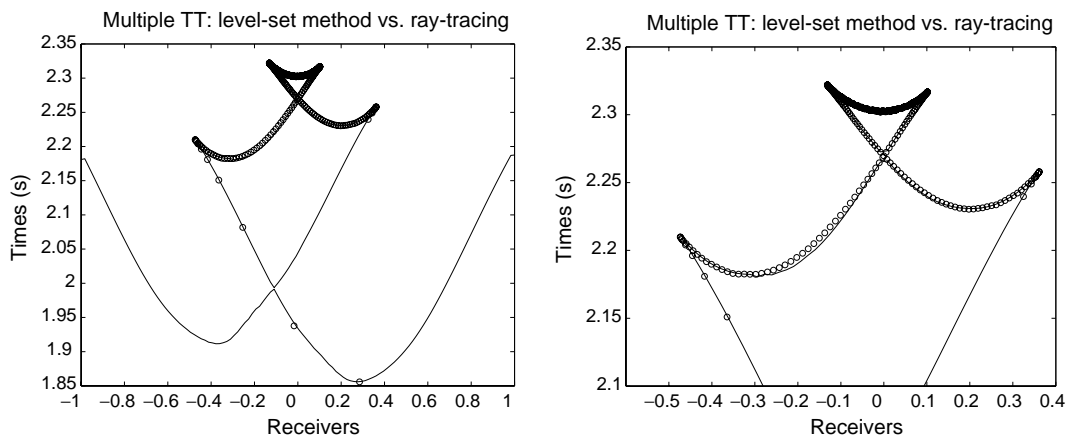


Fig. 7. Eulerian traveltime ('—') computed on a 120×120 mesh vs. Lagrangian traveltime ('○') by a ray tracing method at $z = 2.0$ in Sinusoidal model. The Eulerian approach captures more solutions than the ray tracing method.

finer computational mesh. The computational results are shown in Figs. 11 and 12. As we can see, in this run the level-set Eulerian method yields multivalued traveltimes which match with the ray tracing solution remarkably. From Fig. 12, we can see that the zero level set does have an elongated tip receivers from 260 to 280, and the segments of the zero level-set curve near the tip are indeed very close to each other. Without a finer computational mesh, the level-set method is unable to capture the tip and the related multivalued traveltimes. This demonstrates that given a discretized mesh the resolution is fixed and a method can only capture a finite number of traveltimes.

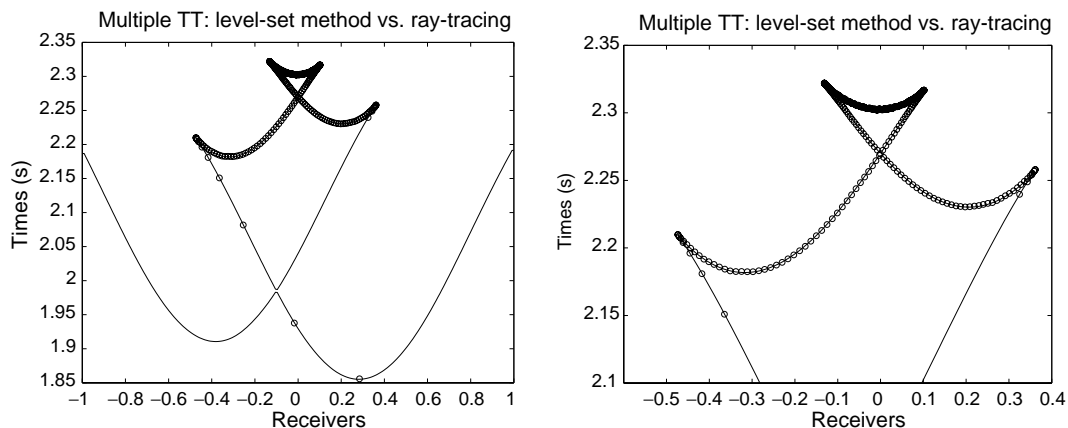


Fig. 8. Eulerian traveltime ('-') computed on a 240×240 mesh vs. Lagrangian traveltime ('O') by a ray tracing method at $z = 2.0$ in Sinusoidal model.

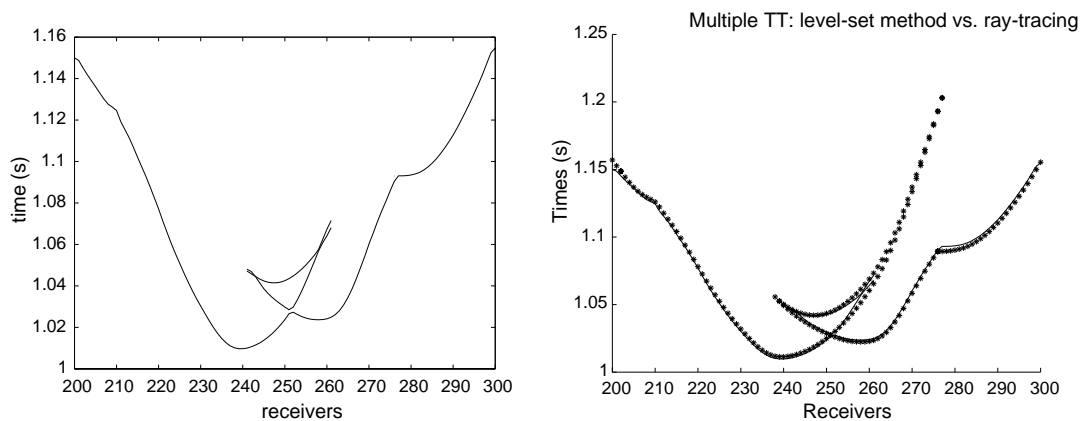


Fig. 9. (Left) Traveltime at $z = 0.0$ km by the level-set method for Marmousi model on a 100×200 grid. (Right) Eulerian traveltimes ('-') vs. ray-tracing traveltimes ('*').

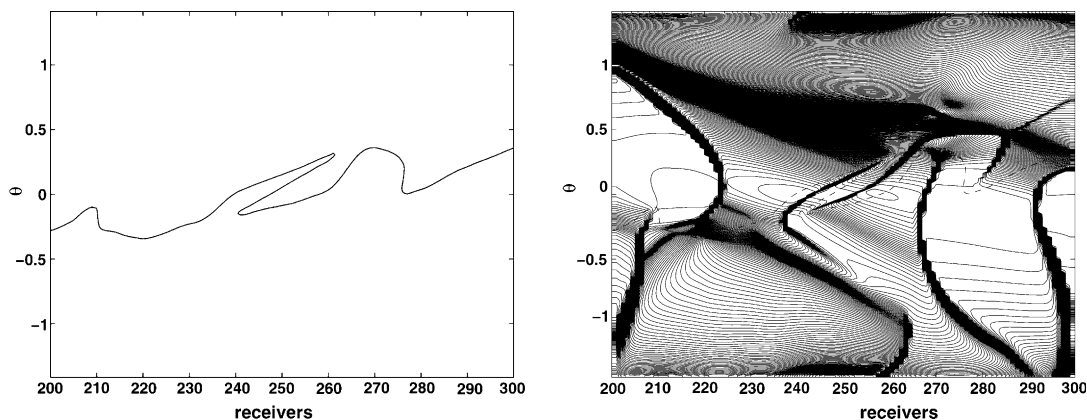


Fig. 10. The zero level set for Marmousi model on a 100×200 grid; the zero level set overlaying contours of T at $z = 0.0$ km.

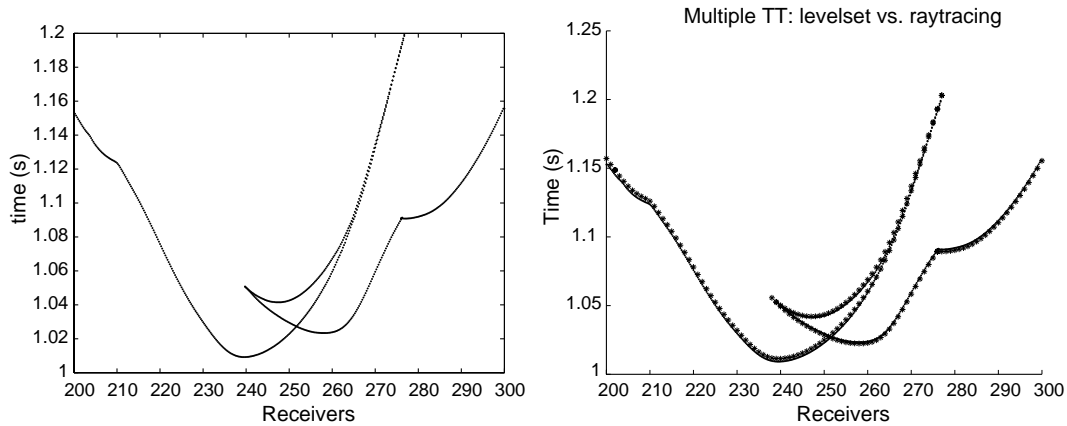


Fig. 11. (Left) Traveltime at $z = 0.0$ km by the level-set method for Marmousi model on a 400×200 grid. (Right) Eulerian traveltimes (‘.’) vs. ray-tracing traveltimes (‘*’).

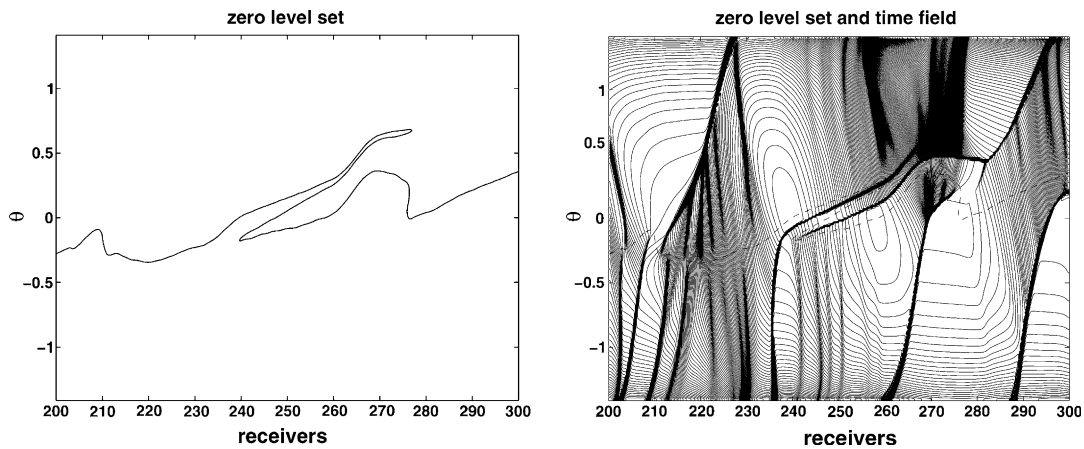


Fig. 12. The zero level set for Marmousi model on a 400×200 grid; the zero level set overlaying contours of T at $z = 0.0$ km.

6.5. An anisotropic model

Although a general anisotropic solid has 21 independent elastic parameters, the transversely isotropic, or TI, solid has only five. It nevertheless retains the essential features of the anisotropic case that we are interested in. Therefore, it is convenient to use TI solids as models to illustrate the advantages of our approach. We consider the simplest case for TI solids, those with vertical symmetry axes, known as VTI solids.

The elastic modulus matrix for transversely isotropic media with vertical symmetry axes has five independent components among a total of twelve non-zero components (see, e.g. [24]). A closed form solution exists in this case for the eigenvalue problem of so-called phase velocities. The quasi-P and quasi-SV slowness surfaces for VTI can be represented as a quartic polynomial equation and the quasi-SH slowness surface can be decoupled from this, leading to the equations [31]

$$c_1 p_1^4 + c_2 p_1^2 p_3^2 + c_3 p_3^4 + c_4 p_1^2 + c_5 p_3^2 + 1 = 0, \quad (55)$$

and

$$\frac{1}{2}(a_{11} - a_{12})p_1^2 + a_{44}p_3^2 = 1, \quad (56)$$

where

$$\begin{aligned} c_1 &\equiv a_{11}a_{44}, \\ c_2 &\equiv a_{11}a_{33} + a_{44}^2 - (a_{13} + a_{44})^2, \\ c_3 &\equiv a_{33}a_{44}, \\ c_4 &\equiv -(a_{11} + a_{44}), \\ c_5 &\equiv -(a_{33} + a_{44}). \end{aligned}$$

In the above equations, a_{ij} are independent elastic parameters of VTI media [24].

Thus, the phase velocities for the three different waves take the form

$$\begin{aligned} V_{\text{qP}}^2 &= \frac{1}{2} \left(-Y_1 + \sqrt{Y_1^2 - 4Y_2} \right), \\ V_{\text{qSV}}^2 &= \frac{1}{2} \left(-Y_1 - \sqrt{Y_1^2 - 4Y_2} \right), \\ V_{\text{SH}}^2 &= \frac{1}{2} (a_{11} - a_{12}) \sin^2 \theta + a_{44} \cos^2 \theta, \end{aligned}$$

where

$$\begin{aligned} Y_1 &= c_4 \sin^2 \theta + c_5 \cos^2 \theta, \\ Y_2 &= c_1 \sin^4 \theta + c_2 \cos^2 \theta \sin^2 \theta + c_3 \cos^4 \theta. \end{aligned}$$

As an example, we compute the three waves for Greenriver shale, which is a typical VTI medium [43]. The five elastic parameters are $a_{11} = 15.0638$, $a_{33} = 10.8373$, $a_{13} = 1.6381$, $a_{44} = 3.1258$, and $a_{12} = 6.5616$.

First, we consider boundary conditions needed in this computation. For the level-set equation we use the non-reflective boundary condition just as done for the isotropic eikonal equation.

For the traveltine equation, there needs to be some extra effort. Because the model is homogeneous and independent of x and z , the velocity component v in the level-set equation is zero. Therefore, there is no need to specify any condition on the boundaries

$$\theta = \frac{\pi}{2} - \epsilon \quad \text{and} \quad \theta = -\frac{\pi}{2} + \epsilon, \quad (57)$$

where ϵ is a small positive number. On the other two boundaries, analogous to the isotropic case, we can invert the characteristics locally and get

$$\frac{\partial T}{\partial x} = p_1 + \frac{p_3}{u}, \quad (58)$$

where u is another component in the velocity field of the level-set equation. With the substitutions of p_1 and p_3 in terms of θ , we have the following differential equations needed to be solved numerically in order to determine the boundary conditions on the two boundaries $x = x_{\min}$ and $x = x_{\max}$,

$$\frac{\partial T}{\partial x} = \frac{1}{V} \left(\sin \theta + \frac{\cos \theta}{u} \right), \quad (59)$$

where V is the phase velocity of the wave mode that we are interested in.

Fig. 13 shows traveltimes for the three different waves computed by the level-set approach (plotted in circles) and comparisons with ray tracing solutions (plotted in solid lines) at depth $z = 0.5$ km. These traveltimes are excited by a point source located at the origin. The upper-left sub-figure in Fig. 13 shows the qP wave traveltime which is the fastest of the three waves. The lower-left sub-figure in Fig. 11 shows the qSH wave traveltime. In particular, the qSV wave (the upper-right sub-figure in Fig. 13) has cusps which imply the multivaluedness at some locations; those multivalued solutions are captured very well by the level-set method. The lower-right sub-figure shows the three waves together.

6.6. Wave guide model revisited

According to the sub-horizontal condition, rays making an angle with the z -axis, θ , greater than $\pi/2$ are ignored in our formulation; no overturning rays in the physical space are allowed. Figs. 14 and 15 show how the method behaves when the sub-horizontal condition is not satisfied.

Figs. 14 and 15 are the solutions from the wave guide model as in Section 6.2 at $z = 2.4$ and 2.8 , respectively. Comparing the first sub-figure in Fig. 14 with the last sub-figure in Fig. 3, we see that the two branches near θ_{\max} and $-\theta_{\max}$ in Fig. 3 are expanding to the outside of the computational domain. Once a part of the branch leaves the computational domain which is seen in Fig. 14, the information carried by that

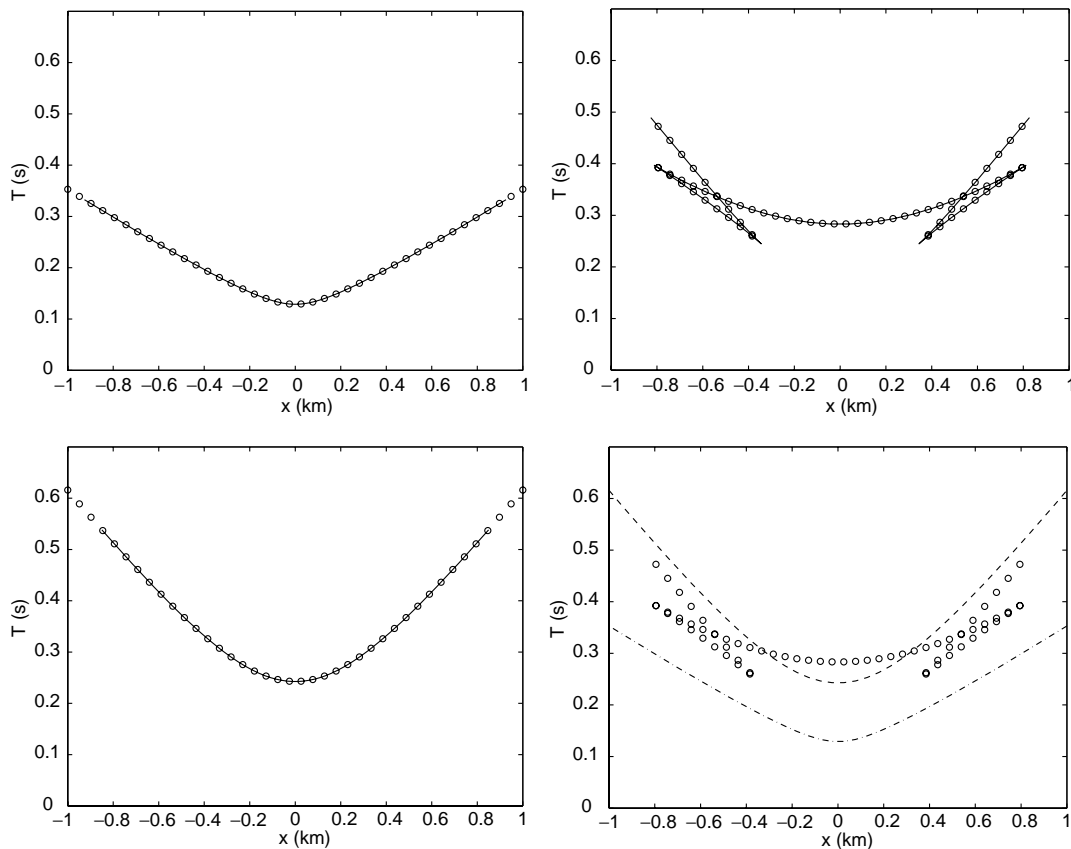


Fig. 13. Anisotropic paraxial multivalued traveltimes by the level-set method. qP wave traveltime, the upper-left one; qSV traveltime, the upper-right one; qSH traveltime, the lower-left one; qP–qSV–qSH, the lower-right one.

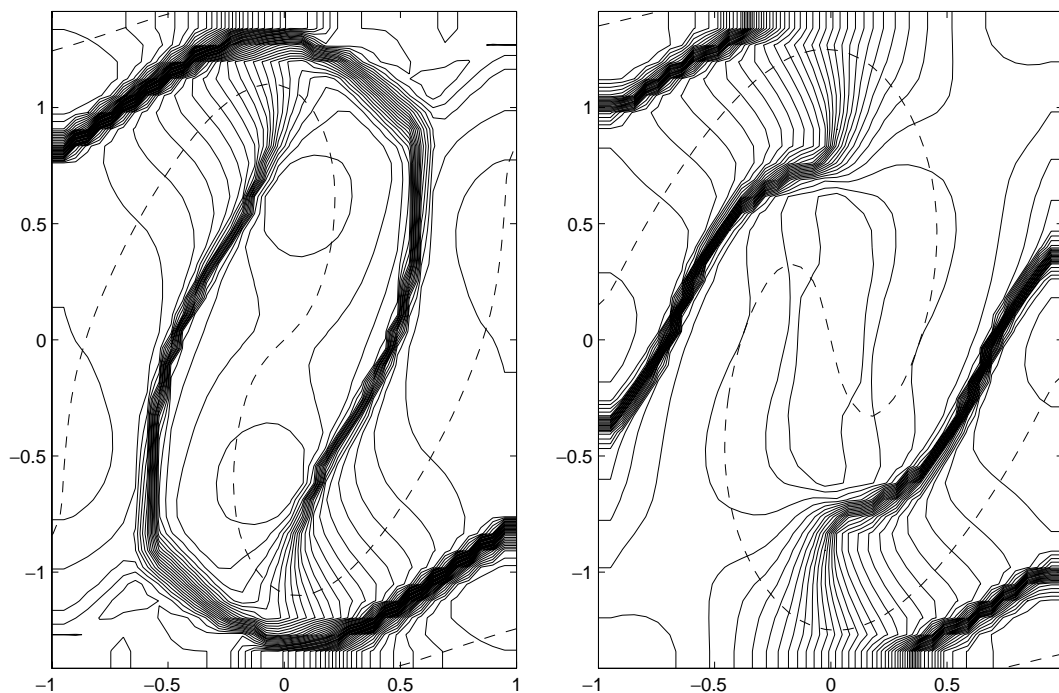


Fig. 14. The zero level-set overlaid on contours of time field T in the wave guide model at $z = 2.4$ and 2.8 .

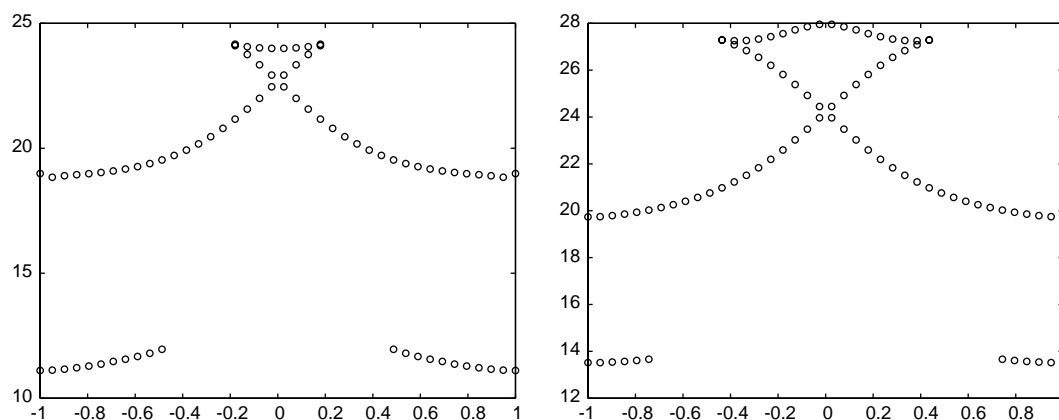


Fig. 15. Multivalued traveltimes by the level-set method for $z = 2.4$ and 2.8 in the wave guide model.

part will be lost; this implies that the related traveltimes can not be recovered. This explains why there are missing pieces in Fig. 15.

7. Conclusions and outlook

We have applied the level-set methodology to compute multivalued traveltimes in the paraxial formulation for both 2D isotropic and anisotropic eikonal equations. The complexity of the proposed Eulerian

method is $O(N^3)$ in the average case and $O(N^3 \log N)$ in the worst case. Numerical examples including the synthetic Marmousi model have demonstrated the accuracy and efficiency of the approach.

Although the formulation presented here is mainly for the point source condition, it will be valid for plane wave propagation too as long as the wave satisfies the sub-horizontal condition; however, we will not present examples here.

Three-dimensional generalization is straightforward in principle, though some numerical issues related to the root finding or the intersection of high-dimensional surfaces are to be settled.

Future work includes computing the amplitude related to the multivalued traveltimes and implementing a localized level-set method [30] which will reduce the computational cost dramatically [32].

Acknowledgements

J. Qian thanks Professor W.W. Symes for many fruitful discussions. J. Qian also thanks Profs. B. Cockburn, S. Osher and F. Reitich for discussions related to this topic. J. Qian and S. Leung were supported by ONR Grant No. N00014-02-1-0720.

References

- [1] J.-D. Benamou, Direct solution of multi-valued phase-space solutions for Hamilton–Jacobi equations, *Commun. Pure Appl. Math.* 52 (1999) 1443–1475.
- [2] J.D. Benamou, An introduction to Eulerian geometrical optics (1992–2002), *J. Sci. Comput.* 19 (2003) 63–93.
- [3] J.-D. Benamou, I. Sollic, A Eulerian method for capturing caustics, *J. Comput. Phys.* 162 (2000) 132–163.
- [4] Y. Brenier, L. Corrias, A kinetic formulations for multibranch entropy solutions of scalar conservation laws, *Annu. I.H.P. Nonlinear Anal.* 15 (1998) 169–190.
- [5] P. Burchard, L.-T. Cheng, B. Merriman, S. Osher, Motion of curves in three spatial dimensions using a level set approach, *J. Comput. Phys.* 170 (2) (2001) 720–741.
- [6] L.-T. Cheng, H. Liu, S.J. Osher, High frequency wave propagation in Schrodinger equations using the level set method, 2003, preprint. Available from < www.levelset.com >.
- [7] L.-T. Cheng, S.J. Osher, J. Qian, Level set based Eulerian methods for multivalued traveltimes in both isotropic and anisotropic media. In: 73rd Annu. Int. Mtg., Soc. Expl. Geophys., Expanded abstracts, pp. 1801–1804. Soc. Expl. Geophys., Tulsa, OK, 2003.
- [8] J.F. Claerbout, *Fundamentals of Geophysical Data Processing*, McGraw-Hill, New York, 1976.
- [9] R. Courant, D. Hilbert, in: *Methods of Mathematical Physics*, vol. II, Wiley, New York, 1962.
- [10] M.G. Crandall, L.C. Evans, P.L. Lions, Some properties of viscosity solutions of Hamilton–Jacobi equations, *Trans. Am. Math. Soc.* 282 (1984) 487–502.
- [11] M.G. Crandall, P.L. Lions, Two approximations of solutions of Hamilton–Jacobi equations, *Math. Comput.* 43 (1984) 1–19.
- [12] B. Engquist, O. Runborg, Computational high frequency wave propagation, in: *Acta Numerica*, Cambridge University Press, Cambridge, 2003, pp. 1–86.
- [13] B. Engquist, O. Runborg, A.-K. Tornberg, High frequency wave propagation by the segment projection method, *J. Comput. Phys.* 178 (2002) 373–390.
- [14] S. Fomel, J. Sethian, Fast phase space computation of multiple traveltimes, *Proc. Natl. Acad. Sci. USA* 99 (2002) 7329–7334.
- [15] S. Geoltrain, J. Brac, Can we image complex structures with first-arrival traveltimes, *Geophysics* 58 (1993) 564–575.
- [16] L. Gosse, Using K-branch entropy solutions for multivalued geometric optics computations, *J. Comput. Phys.* 180 (2002) 155–182.
- [17] S. Gray, W. May, Kirchhoff migration using eikonal equation traveltimes, *Geophysics* 59 (1994) 810–817.
- [18] T.Y. Hou, Z. Li, S.J. Osher, H.-K. Zhao, A hybrid method for moving interface problems with applications to the hele-shaw flows, *J. Comput. Phys.* 134 (1997) 236–252.
- [19] G.S. Jiang, D. Peng, Weighted ENO schemes for Hamilton–Jacobi equations, *SIAM J. Sci. Comput.* 21 (2000) 2126–2143.
- [20] S. Jin, S. Osher, A level set method for the computation of multivalued solutions to quasi-linear hyperbolic PDEs and Hamilton–Jacobi equations, 2003, preprint. Available from < www.levelset.com >.
- [21] P.L. Lions, *Generalized Solutions of Hamilton–Jacobi Equations*, Pitman Advanced Publishing Program, London, 1982.
- [22] Z. Liu, N. Bleistein, Migration velocity analysis: theory and an iterative algorithm, *Geophysics* 60 (1995) 142–153.

- [23] V.P. Maslov, M.V. Fedoriuk, *Semi-classical Approximation in Quantum Mechanics*, D. Reidel Publishing Company, Dordrecht, 1981.
- [24] M.J.P. Musgrave, *Crystal Acoustics*, Holden-Day, San Francisco, CA, 1970.
- [25] S. Operto, S. Xu, G. Lambare, Can we image quantitatively complex models with rays, *Geophysics* 65 (2000) 1223–1238.
- [26] S. Osher, L.-T. Cheng, M. Kang, H. Shim, Y.-H. Tsai, Geometrical optics in a phase space based level set and Eulerian framework, *J. Comput. Phys.* 179 (2002) 622–648.
- [27] S. Osher, R.P. Fedkiw, *The Level Set Method and Dynamic Implicit Surfaces*, Springer, New York, 2002.
- [28] S.J. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* 79 (1988) 12–49.
- [29] S.J. Osher, C.W. Shu, High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations, *SIAM J. Numer. Anal.* 28 (1991) 907–922.
- [30] D. Peng, B. Merriman, S. Osher, H.K. Zhao, M. Kang, A pde-based fast local level set method, *J. Comput. Phys.* 155 (1999) 410–438.
- [31] J. Qian, L.-T. Cheng, S.J. Osher, A level set based Eulerian approach for anisotropic wave propagations, *Wave Motion* 37 (2003) 365–379.
- [32] J. Qian, S. Leung, A local level method for paraxial multivalued geometric optics, *SIAM J. Sci. Comput.* (submitted); UCLA CAM report 03–60, 2003.
- [33] J. Qian, W.W. Symes, Paraxial eikonal solvers for anisotropic quasi-P traveltimes, *J. Comput. Phys.* 173 (2001) 1–23.
- [34] J. Qian, W.W. Symes, Adaptive finite difference method for traveltime and amplitude, *Geophysics* 67 (2002) 167–176.
- [35] J. Qian, W.W. Symes, Finite-difference quasi-P traveltimes for anisotropic media, *Geophysics* 67 (2002) 147–155.
- [36] E. Rouy, A. Tourin, A viscosity solutions approach to shape-from-shading, *SIAM J. Numer. Anal.* 29 (1992) 867–884.
- [37] S.J. Ruuth, B. Merriman, S.J. Osher, A fixed grid method for capturing the motion of self-intersecting interfaces and related PDEs, *J. Comput. Phys.* 151 (1999) 836–861.
- [38] J. Steinhoff, M. Fan, L. Wang, A new Eulerian method for the computation of propagating short acoustic and electromagnetic pulses, *J. Comput. Phys.* 157 (2000) 683–706.
- [39] M. Sussman, P. Smereka, S.J. Osher, A level set approach for computing solutions to incompressible two-phase flows, *J. Comput. Phys.* 114 (1994) 146–159.
- [40] W.W. Symes, Mathematics of reflection seismology. In: *Annual Report, The Rice Inversion Project*, Rice University, 1995. Available from <<http://www.trip.caam.rice.edu/>>.
- [41] W.W. Symes, A slowness matching finite difference method for traveltimes beyond transmission caustics. In: *68th Annu. Int. Mtg., Soc. Expl. Geophys.*, Expanded abstracts, pp. 1945–1948. *Soc. Expl. Geophys.*, 1998.
- [42] W.W. Symes, J. Qian, A slowness matching Eulerian method for multivalued solutions of eikonal equations, *J. Sci. Comput.* 19 (2003) 501–526.
- [43] L. Thomsen, Weak elastic anisotropy, *Geophysics* 51 (1986) 1954–1966.
- [44] V. Vinje, E. Iversen, H. Gjystdal, Traveltime and amplitude estimation using wavefront construction, *Geophysics* 58 (1993) 1157–1166.
- [45] B.S. White, The stochastic caustic, *SIAM J. Appl. Math.* 44 (1984) 127–149.